

SINCLAIR

JULY 1991 £1.95



2 4 70 7
9 10
12 15
16
19
23
34

64

WORLD

*Hardware
review*

**Real Time
Digitizer**

QL IN THE

1990s!

**HELP
us plan
for the
future –**

**SEE OUR
GIANT
JOINT
QUESTIONNAIRE**

*New
User
Guide*



SINCLAIR



Editor
Helen Armstrong

Production Controller
Jayne Penfold

Designer
Jeff Gurney

Advertising Manager
Jason Newman

Magazine Services
Yvonne Taylor

Advertising Production
Michelle Evans

Group Advertising Manager
Jean Dorza

Group Editor
John Taylor

Deputy Managing Director
Ray Lewis

Managing Director
Peter Welham

Sinclair QL World
Panini House
116-120 Goswell Road
London EC1V 7QD
Telephone 071-490 7161
ISSN 026806X

Unfortunately, we are no longer able to answer enquiries made by telephone. If you have any comments or difficulties, please write the The Editor, Open Channel, Trouble Shooter, or Psion Solutions. We will do our best to deal with your problem in the magazine, though we cannot guarantee individual replies.

Back issues are available from the publisher price £2 U.K., £2.75 Europe. Overseas rates on request.

Published by Maxwell Specialist Magazines, A Division of MCPC Ltd., Sinclair QL World is distributed by IPC Market force, King's Reach Tower, Stamford Street, London SE1 9LS. Subscription information from: MSM Subscription Dept, Lazahold Ltd., PO Box 10, Roper St, Pallion Ind. Est., Sunderland SR4 4SN. Tel: 091 510 2290 UK: £21.00, Europe: £32.00, Rest of the World: £38.00.

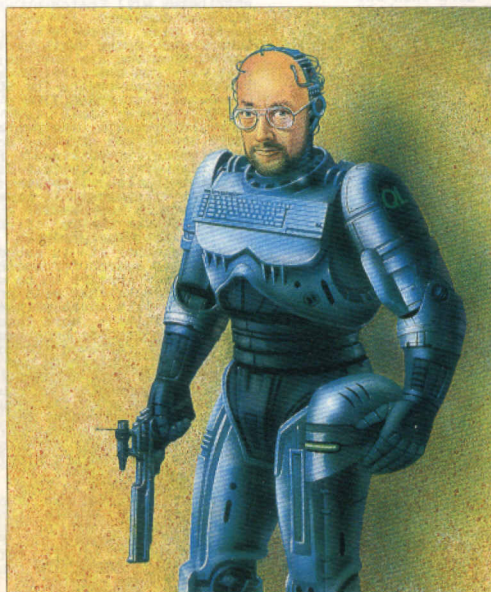
Typesetting by Ford Graphics, 8-10 Whitsbury Road, Fordingbridge, Hampshire. SP6 1BR. Tel: (0425) 655657. Printing by Cradley Print. (0384) 411411. Sinclair QL World is published on the third Thursday preceding cover date.

© COPYRIGHT
SINCLAIR QL WORLD - 1991

C O N T E N T S

■ ■ JULY 1991

- 9 QL SCENE ● A new special interest group**
- 10 OPEN CHANNEL ● Goodbye Quill, hello Perfection**
- 12 QL HARDWARE REVIEW ● CQVI Real Time Digitiser**
- 14 SHOW REPORT ● The Leyland Computer show**
- 16 TROUBLESHOOTER ● A new phase for the QL**
- 19 QL SCENE ● Dutch show a success**
- 23 THE NEW QL USER GUIDE ● Part 5 – input sources**
- 25 QUESTIONNAIRE ● QLAW in cooperation with QL World**
- 31 BE RECONCILED ● Personal financial Abacus tutor**
- 34 DIY TOOLKIT ● Direct access with TRAP keywords**
- 39 SOFTWARE FILE ● Sector X**
- 40 GROUP REPORT ● Quanta in East Anglia**
- 42 SOFTWARE FILE ● Professional Publisher Toolbox**
- 44 LEARN TO LOVE YOUR PRINTER ● Part 2**
- 48 MICRODRIVE EXCHANGE**
- 49 EDITOR'S NOTICE BOARD**
- 50 INSTANT ACCESS**
- BACK COVER – SUBSCRIPTION INFORMATION**



NEXT MONTH

ALTERNATIVES TO ARCHIVE

Mike Edwards assesses Flashback and DataDesign from a personal point of few

DBQL

We continue with Tom Ashcroft's simple relational database.

QL SCENE

A new QL Special Interest Group

This month, *QL World* is co-operating with newly-formed QL special interest group QLAW (QL Advancement Working Group) to issue the first broad-based questionnaire on the QL and its users since the early years of its development (see pages 25-28). This pull-out questionnaire will be analysed by QLAW to provide a detailed profile of the QL community and its future needs. An article or articles based on the analysis will appear in *QL World* later in the year.

Founder Dr. Sohail Bhatti describes the idea behind QLAW: "QLAW is a new interest group formed to encourage the development of QL/QDOS related products, both software and hardware, as a means to the production of a new SuperQL. This new machine is its main goal, and it seeks anyone who would be interested in participating. This includes not only those with hardware and software expertise, but also those who would wish to purchase such a machine. Joining would be a

clear indication of interest, with the expectation that QLAW would use the group strength to generate a QL-compatible machine fit for the 1990s. By sharing resources, QLAW would seek to make the task manageable."

To join QLAW, there is no compulsory subscription, but as launch capital is minimal, a donation of £10 is requested for registration on the mailing list. "I have chosen a sum which I hope will allow such developments as seminars, hardware and software design as well as pay for mailing," says Dr. Bhatti. Users who are interested in joining but cannot afford the £10 might like to approach Dr. Bhatti with alternative suggestions.

A professional statistician and long-standing member of Quanta, Dr. Bhatti has gained support from some fellow-members, members of the committee, and also from QL designers and suppliers. QLAW is however intended primarily to co-ordinate the requirements of users rather than suppliers.

"QLAW is intended as a users' group and so will exclude all traders from influencing QLAW decisions. However, it will seek to develop a consensus amongst all users, so that there are common goals and standards", writes Dr. Bhatti.

Ultimately QLAW's aim is to co-ordinate design skills and user needs within the QL community to develop and extend the QL to meet the challenges of the 90s. He argues what many users believe: that the QL and QDOS is an advanced system which is under-used. Now is the time, he believes, to consider what can be achieved by pooling the best of our resources, and spreading the load of development.

Up to date news will be shared through the mailing list, and designers would be able to call on the membership for beta testing of new products. A non-disclosure agreement would ensure security and a fair share in the gains for individual design work.

The QLAW questionnaire will also appear in a substantially similar form in *Quanta*, and

CGH Services' QL Technical Review, as well as through various overseas user groups. The *QL World* questionnaire incorporates a number of questions relating to the availability and value of *QL World* itself which will not, of course, appear on other drafts of the questionnaire. The analysed data and comments in this section help the magazine for QL users to plot its future contents.

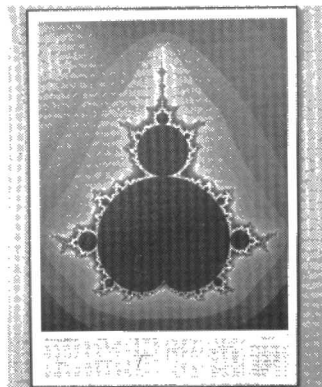
To join QLAW or obtain further information, write to **Dr. Sohail Bhatti, 4 Wasdale Avenue, Park View, Blackburn, Lancs. BB1 1XD**. Please make cheques payable to QLAW. Stamped self-addressed envelopes are appreciated with enquiries, to keep down running costs.

QLAW and QL World have no association other than in the publication and analysis of the QLAW/QL World questionnaire appearing on pages 25-28 of this issue. QLAW is a non profit making user group which does not claim endorsement from any commercial organisation.

Fractal poster

Organised Chaos is a business in Manchester which is dedicated to producing high-quality small-poster sized fractal images. The sample deposited with *QL World* is an A2-sized, glossy poster showing a Mandelbrot image on a purple background, with a short history and commentary on the Mandelbrot Set at the base of the sheet.

Says OC's Colin Menhinick: "The computer generated images are as clear as a colour



photograph. There are no marks, castle edges, blurred spots or fine lines missing. You get the full majesty of the image in hairsbreadth detail." He sees the posters as an educa-

tional resource for Maths and Computing departments, and the sciences in general.

Organised Chaos offers its posters with a money-back satisfaction guarantee. The posters cost £9.95 inclusive of VAT, plus £1.50 post and packing. A block-mounted version, designed for school and college use, with a washable surface, costs £29.50 inclusive of UK carriage and hanging set. The poster sent to us arrived in a heavy-duty card tube with a strong plastic stopper, in perfect condition.

Orders and enquiries to **Organised Chaos, 10 Stanton Gardens, Manchester M20 8PT. Tel: 061 228 7356**.

Guide slip

Our apologies for two misprints in Part Three of the *New User Guide*. At the bottom of page 27, the PAPER command should read PAPER#2,4, and on page 28 under the AT command, the commands in the text should read AT#2,4,2 and PRINT#2. The short program over pp.27 to 28 should finish with:

CLS #2

Reader Dr. D C Lewis of Shrewsbury points out that the reference to the window number at the bottom of page 27 should more properly read 'channel number'.

OPEN CHANNEL

Open Channel is where you have the opportunity to voice your opinions in *Sinclair QL World*. Whether you want to ask for help with a technical problem, provide

somebody with the answer, or just sound off about something which bothers you, write to: Open Channel, *Sinclair QL World*, 116/120 Goswell Road, London EC1V 7QD.

Rejoice

I have been a regular daily business user of *Quill* for the past five years, and am writing this letter as a result of seeing a pre-production version of Digital Precision's *Perfection* demonstrated at the London Quanta sub-group.

As someone who is all too familiar with the destructive quirks of *Quill*, I would say, forget your concerns that nothing can be as useful and as uncomplicated as *Quill*, and put your order in for one now.

I am pleased to say that this is probably one of the last letters I will ever write where there is

a possibility that the cursor will disappear and that in an instant I will lose a carefully worded document.

Nigel Spencer
London SW17

Register

I found Gavin Monk's LDEF command described in June 1990 very useful, and so was disconcerted when on one occasion it wouldn't work with the later procedures and functions in a Basic program. On re-checking the listing, I noticed that in lines 53 and 74 index register A2.W instead of

A2.L was used. This is alright when Basic programs are small, but fails if they are larger, because then the name table and name list come higher in the Basic area than the program.

Also, I am surprised that after line 68 we do not have EXT.W D1 and EXT. L D1 to allow trouble-free addition of D1 to A1 at line 92. After all, we are told that CA. GTSTR corrupts D1 and, at line 41 that seems to be the last place D1 is used. At line 68 only D1.8 is set, so the rest of D1 could be anything.

G D Gwill
Edinburgh

Minerva

Some time back you asked for people's experiences of the *Minerva* rom. Here are some of mine. I've had three versions of *Minerva* V1.61, 1.66 and 1.81, each one better than its predecessor and all of them better than the old JS roms. I use my machine for word processing, using DP's *Editor* and Psion *Quill*. I also run Borland's *Turbo C* under *Conqueror* for a college course.

As to what I like about the Minnie, 90 per cent of the time I don't even know it's there (as should be the case with an operating system). I now no longer have to waste valuable memory loading third party software to speed up graphics and math routines, thought I do use the excellent *Lightning* rom text extensions. My SuperBasic programs run faster (as does the machine generally), and when I make a syntax error *Minerva* shows me the point on the line where I made the mistake.

Not only can I use cursor left/

right to navigate round a line, but *Minerva* implements Home and End keys with an ALT LEFT/ALT RIGHT combination. There are also line delete keys.

My machine starts up faster (I have a full Trump Card), there is no more waiting ages for the ram test. *Minerva* does a more thorough test of ram, and if I have a duff chip, *Minerva* will tell me, so I can now be more confident of my QL hardware. If I run a QView program, I can even get a schematic diagram of where an offending ram chip is.

I like the idea of Qdos being back under development. It is nice to have an operating system that has customer technical support again. Qdos was never 'finished', and anyone who wants to stop its development at the old MGs obviously hasn't grasped Sinclair's original dream for the machine. *Minerva* isn't perfect, but it is getting there.

Dominic Brown
West Kingsdown
Sevenoaks

Strings

I have found a bug in SuperBasic I have not seen detailed before. It concerns the comparison of strings. Some characters are considered equal by the interpreter, which are obviously not. I stumbled on this while writing a guarded input routine; the character 0 is effectively the same as the cursor up key. The only way to distinguish is to compare CODEs. I have since found a total of 64 cases (see enclosed printout). There is an interesting effect: for example, A is equal to CHR\$ 225, and a is

Editor's notebook

Looking back over the QL's history, we see various drives that were made towards an integrated QL, equipped with disk drives and a far larger memory capacity. New machines have sprung up and then faded as their producers fell foul of various obstacles. We strain an ear to hear news of the Thor, but all official sources have been ominously silent – in this quarter – for some time.

Yet 1991 may be the year of regeneration. Miracle Systems have released a major upgrade board. Digital Precision have produced a top-of-the-line word processor. Is this the time for new special interest group to be calling for more co-operation towards developing upgrades for the QL? Hasn't it been done?

I think the answer is yes and yes. It is being done, but further co-operation from people of talent and commitment may be hoped to produce further good. It may fall on stony ground – but why not try?

equal also to CHR\$ 225 but A is not equal to a (using == completes this).

A few months ago, Mike Lloyd gave a function for signum: RETurn 1-(x<=0)-<x<0). This does have an elegant simplicity, however, the alternative, RET <x>0-<x<0> works just as well.

Caleb Leeke
Huntingdon
Cambs

Tonto

For some time I have been using a BT Merlin Tonto computer. I understand that it is a derivative of the Sinclair QL and was on the market in the early 1980s.

I use it both for its word processor (Quill) and its spreadsheet (Abacus), as well as the telephone side, and find it an invaluable office tool.

I have been reading QL World for six months now and find it very interesting, but wonder how relevant the articles are to my Tonto in the long run. What do other Tonto users (if there are any) feel?

L M Westall
Yarnton
Oxford

PC Format

I have vast amounts of data created using Quill, Abacus and Easel. I have to be able to access this data using any of a selection of programs on the PC, including the Psion PC4 package, which looks a lot like the QL version.

I have PC Conqueror and its disk conversion facility, but am unable to access the files correctly once they are on PC format disks. What do I do to get my files into a readable format?

C A Bland
Green Street Green
Orpington

Editor's comment: First, you should ask Digital Precision's advice. Always talk to the publisher first. Also, somebody experienced in using DiscOver may be able to advise. Transferring data between one operating system/utility combination and another totally different combina-

tion can be a knotty problem even with bare Ascii files.

And again

Here is a little problem that I would like some help with; I have been using an IBM compatible (an M31PC SMC PC-386XP) for the past 18 months (along with my trusty QL). The IBM clone has both 3.5in and 5.25in floppy drives, as well as a 40MB hard disk, and runs DR-DOS using all the function keys from F1 to F12.

I will shortly be losing the use of the M31PC – how can I transfer all my software and files to use on my QL? This is a considerable amount. What would I need to purchase to run all this on my QL? And what would it cost? I am using Ability Plus from Migent, Sage Accountant Plus, Logitech Finesse DTP and some other simple programs. Will I be able to use these? My main software packages are stored on hard disk and I use this to run each program using the CD/command. But I have very little actually stored on the hard disk.

Please bear in mind that I am not totally computer literate and just get by with simple instructions.

Just to give you all a chuckle I have yet to add disk drives to my QL and am, like a few others, still running the basic machine with mdvs. I know that I will have to upgrade, but is it going to be financially viable?

John Crompton
Oakehampton
Devon

Editor's comment: the software exists to port material from a PC-compatible to a QL system, and also emulate the MS-DOS environment on a QL. Your first move should be to talk to Digital Precision, who publish the main PC emulation program Conqueror.

You will certainly have to expand your memory if you are to run a DTP package on a QL. Miracle Systems market a disk drive interface and memory expansion board, and EEC Ltd. may also be able to advise. Beware of moving straight to a hard disk if you are not very confident on your computers.

As for which programs you can use, again this is something you will have to talk over with the publishers of the emulator, because it is on that that you will depend.

As to whether it is financially viable, that will depend on cost of setting up your own PC system versus cost of expanding your QL. There are two points you haven't covered in your letter; one is how much you currently use your QL and what investment you have already made in it, in software, and the other is: are you the owner of the PC-compatible programs you list, or are you relying on copying them from your place of work? If you were considering the latter and the legal aspects of this had not occurred to you, remember that you would be in breach of copyright, and also that your programs would then be unable to claim upgrades or seek assistance from the publishers.

If you have to rebuild your software collection rather than simply porting the data, this will make the sums look rather different. There is a great deal of good software for the QL, including accounts and DTP.

Agenda

Am I alone among your readers, or is there another Agenda/QL user out there? The Agenda is an organiser with several extra features compared to those mentioned by David Drysdale last autumn – touch typing entry system, and the most remarkable Finding and Printing system that I have yet to find. But when I came to trying to store backup files on the 3.5in disks of my Sandy-expanded QL(JM rom) using serial interconnection, I found that I need some routines to negate print instructions when sending "organiser to QL", but could send anything the other way.

I use Editor Special Edition, from Digital Precision. The Agenda has a very useful system of sub-files, one for each letter of the alphabet within which searches can be made very quickly; addresses under A, telephone numbers under T are obvious choices. My address list transferred to QL (with a few non-print characters) and my only worry was a lack of indication when transfer was complete. The same file sent

back to Agenda arrives as a series of lines with nothing to show where one address ends and the next one starts.

It is a good to be able to write letters on trains and buses, but the ability to edit on a monitor before printing and store on disk would be a big asset. I write in the hope that someone else has made some progress along the same path: my 24-pin D connector, QL Ser and 25-pin D connector numbers are as follows:

1,2,3,5,7,20

GND, 2 3 4 1 5

-, TX, RX, CTS, Sig, GND, DTR

Alec Wilson
Colden Common
Winchester

Cattle

I bought a QL about five years ago with the idea of using it for my work, which is looking after 150 cows. The idea was to use it for breeding, milk records, health records and the pedigree records of each cow. Because the QL User Guide is so large and it takes time to learn everything about the QL, I have never got as far as using it for my working records.

Now the National Milk Records have come up with Herdfile software. They supply a single disk every month with all the details I require up to date. The problem is that the disks, which are available in 3.5in and 5.25in, are only suitable for IBM or IBM compatible computers. My other question is, can a disk drive of any make be fitted to my QL to enable me to use this service?

C A Trim
Alresford
Hants

Editor's comment: Software designed for IBM personal computers and their compatibles runs under the MS-DOS operating system, not QDOS as used by the QL. To run programs under MS-DOS, you would need an MS-DOS emulator. You should talk to Digital Precision about their program PC Conquest. Not all MS-DOS software will run under a specific emulator, so they may need to see a test disk.

QL HARDWARE REVIEW

CQV1 REAL TIME DIGITISER

The new real time video digitiser from CL Systems arrived in a substantial parcel a couple of weeks ago, and television has not been the same since.

Although they do a similar job, video digitisers and digital scanners are very different pieces of equipment. A digital scanner is an optical device which is moved across a printed image in order to copy the image into a computer's memory. Jurgen Falkenberg's scanner is attached to a standard dot matrix printer head in order to provide motive power. A real-time video digitiser, on the other hand, also converts images into a digital format, but its image source is a video recorder or a video camera. Where a digital scanner can take ten minutes to scan its image, a video scanner has to perform the same task in a fiftieth of a second in order to capture a video frame. The results can be astonishing.

Scanning

Video scanning has a number of practical uses. The computer can be programmed to capture an image every few seconds or minutes in order to produce a time-elapse sequence of images. Point your camcorder towards a cloudy sky (nowhere near the sun, of course) and the QL can be programmed to capture and store an image every five minutes for almost two hours. Replay the images and you will see the same cloudscape unfold in a few seconds.

Alternatively, the video digitiser can be used to print out selected images from a camcorder sequence to assist in editing, or to provide a printed graphic record of what is on a video cassette.

Desk top publishers are going to find an instant use for video digitising because it can capture video and tv screens in a form suitable for importing into documents. Unlike the digital scanner's output, the images produced by the video digitiser need very little 'cleaning up'. Of course, if you use an art program like Eye-Q you can use tv screen images sampled by the digitiser as the basis for your own original artwork.

Firstly, though, the practicalities. The video digitiser is a box 18cm by 11cm by 5.5cm which fits into the rom expansion slot at the rear left of the QL, completely dwarfing the displaced Super Toolkit II

■ Mike Lloyd meets a machine that takes pictures from the tv, and keeps them.

rom which normally lives there on my system. Although it is a large box it feels extremely light. In order for it to be stable it must sit on the same surface as the computer, so the QL's ridiculous plastic feet will have to be removed. If the QL is sitting on a highly polished surface, a bit of Blu-tak under both the computer and the digitiser will ensure that they remain married together throughout a session.

A major logistical exercise, in our house at least, was the positioning of the QL within a cable-length of the video recorder. If you are still using the family tv set for your computing this is unlikely to be a problem, but it's a little more complicated for those of us long since banished to the spare bedroom to practice our black arts. Following successful negotiations, the QL, disk drive, keyboard extension and a six-way power distribution block arrived in the front room. The printer stayed upstairs because the CQV1's software allows images to be saved to disk and printed later.

The sequence of events at the start of a digitising session is critical. Firstly, have everything turned off while the cable connections are made. Set up the QL system as you would normally and add the video digitiser to the rom port. Next, connect the digitiser to the video source (a camera

or a video recorder) with a standard phono-plug cable such as those found on stereo systems and usually with camcorder systems.

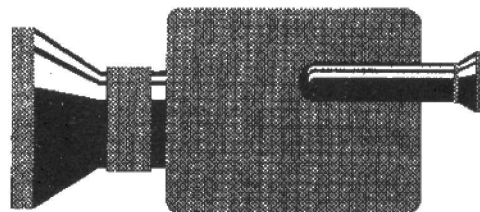
After making sure that all the connections are sound, turn on the video source before booting up the QL. This ensures that no damage is done by stray electrical surges and imbalances between the various circuits. If the QL does not boot in the normal way, remove the power as quickly as you can. Incidentally, the instructions provided with the review copy of the machine tell you to turn the video recorder on first some two paragraphs after it tells you to turn the QL on.

Samples

Two boot files are included on the disk accompanying the digitiser, one for floppy disk systems and another for microdrive users. Other files contain a small amount of runtime extensions and the main executable file. A couple of sample screens are thrown in for good measure.

The default system boot file simply loads the extension codes, leaving you to launch the executable program in your own time. This is sensible because the digitiser is fully multi-tasking, allowing you, for instance, to have a real-time tv image show-

QL Video Digitizer Main Commands



F1 Draw Menu
F2 Save Image
F3 Load Image
F4 Sample Input
F5 Clear Screen

D Select Display
S Auto/Manual
sample mode
N Negative Image
I Inverted Image
P Print Dump

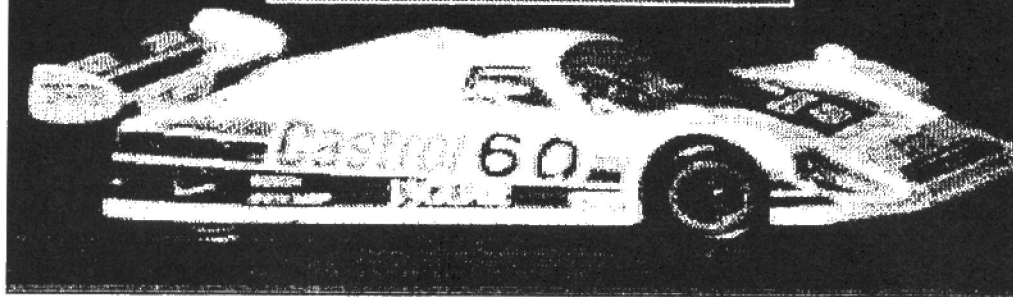
SHIFT +
F2 Store Image
F3 Recall Image

CTRL +
F3 Merge Images

CTRL/SHIFT +
F3 Merge Images

QL REAL TIME VIDEO DIGITIZER

BY CL SYSTEMS



Some sample screens included by CVS with the video digitiser package.

ing in the top corner of a screen otherwise devoted to a Quill session.

When the program has been loaded the main menu appears below a digitised picture of a sports car. The controls are very straightforward and can be memorised within a few minutes of using the system. The function keys are used to display the menu, control disk/micro-drive access, start the real-time sampling and clear the screen. Throughout the review period the software behaved faultlessly, like the construction of the device itself, gave the impression that the video digitiser is a very well thought out peripheral produced by a company with high standards of workmanship.

Capturing

Pressing F4 starts the image capturing process in manual mode. Every time the space bar is pressed a video image appears on the screen. To move to the real-time sampling mode the 'S' key is pressed. With the full QL screen being used to display the images a new frame is produced every two and a half seconds. The clarity of the image is amazing, especially as the QL has only eight colours and a 256x256 pixel resolution at its disposal.

It is probable that at the beginning of each sampling session the digitiser's contrast will need adjusting. This is done by turning the only control the digitiser needs, a small red knob mounted at the back of the box. At one extreme, the QL display is all whites, cyans and yellows while at the other the picture is black, red and blue. Optimum adjustment is easy to achieve. This facility contrasts with the post-scanning adjustment of the colour balance available on Falkenberg's digital scanner.

Three screen modes can be selected whether you are in 'step-mode' or are sampling 'real-time'. Pressing 'D' toggles between a full-screen display, a quarter-screen display and four quarter-screens refreshed sequentially. The quarter-screen display in step-mode proved extremely interesting during action sequences in films. Quarter-screens are refreshed so quickly that the refresh rate does not be-



come obvious except during camera-panning sequences.

The clarity of the images matched with the sometimes bizarre colouration turns even the duller film into a new art-form. Printing out a picture on a dot matrix printer, however, turns the colours into a more acceptable grey scale. The print quality provided by the in-built print dump routine was quite sufficient for normal use, although I found Sidewinder was invaluable for cropping and re-sizing the images.

Video tricks

Some additional video tricks have been added to the digitiser package. Screens can be stored in ram by pressing Shift-F2. With expansion fitted, more than a dozen full-screen images can be saved. Other keypresses allow you to recall the pictures, invert them, turn them into 'negatives' and merge them into each other. Their value is perhaps open to question,

but the code to implement them is so simple that they do not add much to the bulk of the controlling program. For more involved manipulation of the images Eye-Q permits greater expression.

The CQV1 video digitiser proved easy to live with, once the computer and the video had been moved to within sight of each other. The menu selections were sensible and the more commonly-used ones were readily remembered. The program operated flawlessly and produced no clashes with other programs tested. The digitiser is available from CL Systems (See also their ads in the back of QL World) for £123.000

INFORMATION

Product: CQV1 real-time video digitiser.

Supplier: CL Systems, 403 Chapter Road, Dollis Hill, London NW2 5NG. Tel: 081 459 1351.

Price: £123, inclusive, state mdv or flp.

Miracle Systems' new Gold Card provided all the buzz at the April 27 Leyland computer show organised at Runshaw College, Leyland, Lancs, by Sector Software's David Batty.

As some enthusiasts prattled on about the Gold Card heralding the new golden age of the QL, others wondered just what they would be getting for £375, which, let's face it, is almost as much as some people paid for the computer in the first place.

The Gold Card, it seems, is a replacement for all existing ram-plus-disk interfaces and other QL expansions, and a trade-in allowance will be made against them varying between £25 and £150.

The new expansion will speed up the QL by as much as four times and give almost two Megabytes of ram. It will also accept new disc drives with a total capacity of up to 3.2 megabytes. In addition the expansion houses a battery-backed clock.

A battery-backed clock is also a feature of the new £65 Minerva Mk II rom, introduced at the show by Tony Firshman. This clock, on its own chip with 256 bytes of ram, has the advantage of being completely crashproof. The new Minerva MKII has a Philips 12C interface which is able to drive a whole range of chips. It also has a switching facility as part of an analogue/digital Philips chip interfacing to the 12C bus. This facility was used, at the show, to control a robot playing the old Chinese game of 'Towers of Hanoi'. The battery ram can also be used to autoboot QLs, via the network, without any drives being fitted.

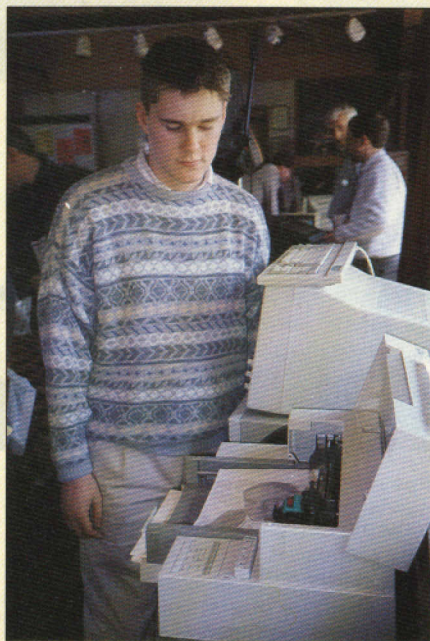
Another new product was Digital Precision's word processor *Perfection* which Freddy Vachha says 'will knock your socks off'. How many programs Freddy actually sold at the show is anybody's guess, as the demo was so fast and furious that it was difficult to get even the tiniest order in edgeways! Plenty of leaflets were being picked up, however, and there is no doubt that orders will come rolling in by post once the many features of this idiot-proof word processor have been digested.

NEWS FROM THE NORTH

David Drysdale visits the show at Leyland, and meets the newest and best.

This after-the-show sales boost is par for the course on these occasions according to Tony Phipps, Business Development Manager of EEC Ltd., whose stand had leaflets offering an Epson 3.5in 720KB floppy disk drive for just £38 including VAT. A Sinclair 14in CGA colour monitor was also on special offer at £120 and back-up JM version QLs for only £85. Tony says he is highly optimistic about the future of the QL. His company, he says, will continue to support it to the full.

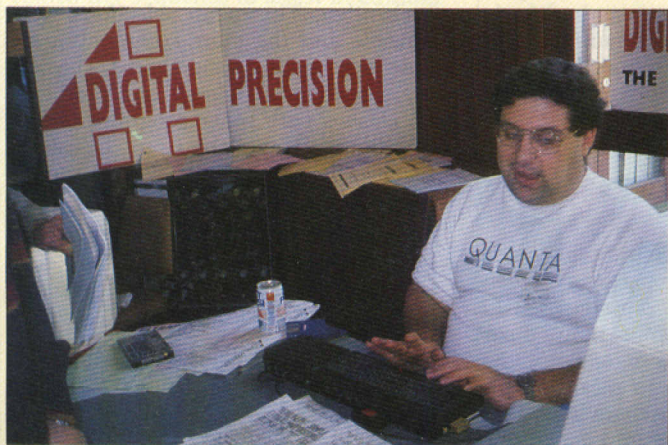
The show's organiser, David Batty of Sector Software, was finding plenty of interest for his £14 crossword-solving program *Wordsmith* which could find words and solve anagrams at lightning speed. He demonstrated by keying the letters 'ology' and thirteen words containing it came up in less than one and a half seconds. His *STD-Index* program was also selling well. This stand-alone database of over 6000 UK dialling codes was developed to enable users to find a place location from a telephone number. People buying goods advertised in magazines



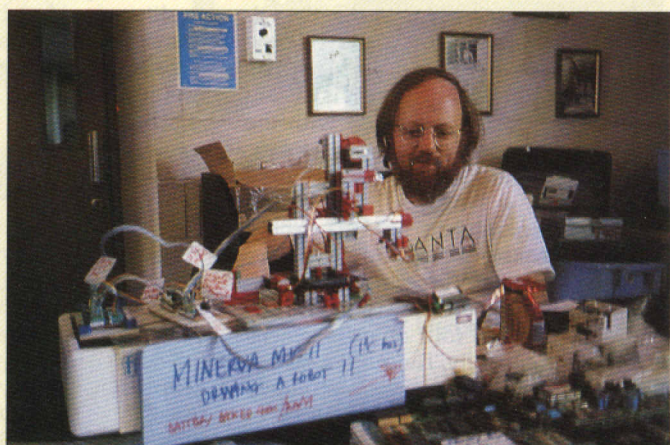
QLAS printing SPEM portraits.



Minerva author Lawrence Reeves.



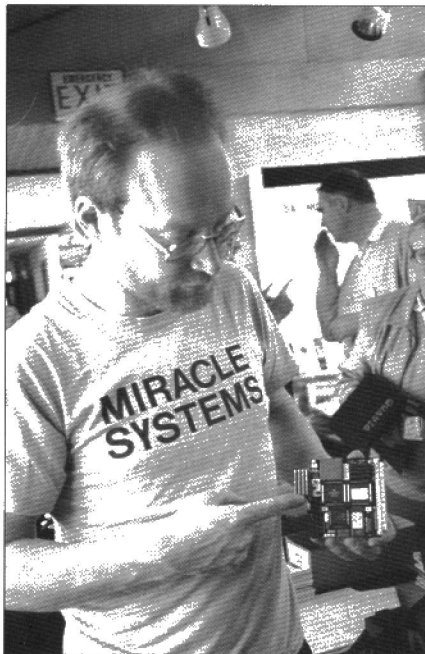
Freddy Vaccha of Digital Precision.



Tony Firsham demonstrates the new Minerva.



Tony Phipps of EEC Ltd., and stock.



Miracle Systems' new Gold Card.

such as *Exchange and Mart* would find this program useful as they would know how far away the vendor was and whether the bargain was worth picking up. With director enquiries costing 40p a time this program, at £14, would pay for itself in the first 35 queries.

Other inexpensive programs which seemed to be selling well were *QL Genealogist* and *Super Disc Labeller* on the Dilwyn Jones Computing stand. At £19.50 Genealogist was his best selling mail order program, said Dilwyn, but a lot of interest was now being shown in Super

Disk Labeller which not only produced neat labels in various type sizes but also printed disc sleeve inserts and Directory page printouts. The cost of this program was only £10.

A tightly packed and up-to-date 16 page list of programs including games, utilities and fractal and mathematical recreation programs was available on the CGH Services stand. One program for sale soon is *Cribbage and Bridge Scorer* which will allow the well-known game of crib to be played on screen.

Club secretary Ron Dunnnett and Dennis Briggs were representing Quanta at the show. For Ron it meant an exhausting 500 mile round trip. The local Quanta Lancs Area Sub-Group QLAS were there too displaying their newly minted tin badges bearing the QLAS logo. Their stand featured a video camera and Spem digitiser, which they were using to make portraits of the visitors.

Although it was an all-formats show, the QL sector was dominant. Other exhibitors seemed pleased, however. One non-QL dealer, Mr S A Raja of Scan International, told me he had sold over 3000 3.5 floppy disks during the day, on top of printer and computer sales. I left the last word to show visitor Richard Cooke of Wigan. Did he intend to buy anything at the show? "Of course", he replied. "The new Gold Card, what else!"

CARE ELECTRONICS

QL SUPERTOOL KIT II by Tony Tebby

Over 118 Commands:- Full Screen Editor, Key Define Print Using, Last Line Recall, Altkey, Job Control, File Handling, Default Directories, Extended Network, 16K Eprom Cartridge Version@£23.50d
Configurable Version on Microdrive@£23.50d

MIRACLE SYSTEMS PRODUCTS

OK Disc Interface (Inc Tool Kit II)@£99.64b
QL Centronics Printer Interface@£29.61d
QL Expanderam 512K ThruCard@£99.64b

QL HARDWARE

Single 3.5" Disc Drive & (Own PSU)@£105.75a
Dual 3.5" Disc Drive & (Own PSU)@£176.25a
3.5" DS/DD Discs 10 off@£9.40c
QL Keyboard Membrane@£11.75d
Care Eprom Cartridges each@£6.11c
ULA Chip ZX8301@£15.98c

SOFTWARE 87 (State MDV or Disc)

TEXT87V3.00 (Requires Min 256k)@£60.00d
FOUNTED 89@£15.00c
FOUNTEXT 88@£25.00c
TEXT 87/FOUNTED 89/FOUNTEXT 88@£94.94b
2488 PRINTER DRIVER@£15.00c
Upgrade to Text 87 V.3.00. Return old copy together with@£25.00d

MONITORS (Price including lead)

Philips CM8833 Colour Med-Res@£258.50a
Philips BM7502 Green Hi-Res£99.64a

HOW TO ORDER:

By Post. Enclose your Cheque/PO made payable to CARE Electronics or use ACCESS/VISA. Allow 7 days for delivery

QPAC II

QPAC II Main menu windows adjust automatically to size. Files: directory, view, print, delete, backup, jobs, pick, Rjob, sort, channels, things exec, wake, buttons, Hotkey. Hotjobs. Fully multitasking, allows many programs to run at once. Requires min of 256k expanded memory@£49.35b
Upgrade QRAM to QPAC II return master@£29.61b

PLEASE SEND SAE FOR FURTHER INFORMATION

TONY TEBBY SOFTWARE (QJUMP)

QLFP (Micro/P disk interface upgrade)@£15.51d
QTYT Type/Spell Checker@£30.55d
QPAC 1 Desk top accessories, calander, alarm, calculator, typewriter, digital clock sysmon@£22.56d

ZITASOFT SOFTWARE by Steve Jones

LOCKSMITHE copies M/DRIVE - M/DRIVE@£14.10c
4MATTER + LOCKSMITHE copies M/DRIVE - DISC@£23.50c
SIDEWINDER - High resolution printer driver prints full screens or parts of screens from postage stamp size to large banners, Prints sideways, invert, scale, mirror, text insertion@£20.21c
SIDEWINDER PLUS - (for expanded QL's) includes all the features of above, PLUS multiple label printing, desktop publishing files and printer driver for 24" pin plus printers LC10 and Jx80 colour printers. (Please state 3.5" disc or M/D)@£23.50c
Upgrade to Sidewinder Plus@£8.46c

QPOWER REGULATOR

Switch mode power supply, QL runs cool, stops, lock ups (due to voltage drop). Helps filter noise out. Simple to fit (no soldering) £25.85c

MULTI COLOURED PRINTER RIBBONS

Huge range of coloured printer ribbons available in RED, GREEN, BLUE, BROWN, PURPLE, YELLOW, ORANGE, MAGENTA, SILVER and GOLD
Prices from as low as £3.00. Please phone for price and delivery.

READY MADE LEADS

RGB QL TO PHONO£6.11c
RGB 8-6 PIN£7.52c
RGB 8-7 PIN HITACHI£7.52c
RGB 8-7 PIN FERGUSON£7.52c
RGB 8 PIN TO SCART£11.75c
6WAY PCC TO 25WAY 'D'£9.87c

24 Hour Order Line 0923 894064 OPEN

9am-5pm Mon-Thu
9am-4pm Fri

CARE ELECTRONICS

DEPT QL, 15 HOLLAND GDNS
GARSTON, WATFORD,
HERTS, WD2 6JN.
Tel: 0923-894064
Fax: 0923-894064

Please add carriage
a=£11.75 b=£3.76
c=No charge d=£2.35

T A P R O U B L E

Every month or so, there is some software and/or hardware development (however minor it may seem) on the QL scene. The *Gold Card* has been further developed, and has been demonstrated with both 12 and 16 MHz cpu chips. It now looks as though the 16 MHz version will be available right from the start, and the small difference in price suggests there is little point in buying the slower version.

Considerable improvement has been shown in performance of various programs, when run under the Gold Card. *Quill* clearly runs much faster. *Perfection* is even faster than before, but it was so fast anyway that the improvement is in fractions of seconds. Even *Lightning* has more effect. *Professional Publisher* is a program which shows improvements in some areas but not in others; routines that are written to run in a set time will still do so, but menus appear much faster. The relative slowness of menu activity irked me as much as anything it should be nicer to use now.

Perfection continues to change its face, although the 'works' have been mostly finished for some time. The user interface is reported to be much improved and a definitive version of the program should be available for checking any day now. There is no doubt that the combination of the Gold Card, high-density disk drives, and *Perfection* will transform the QL for word processing work.

QFile reports that *Discopy* has reached version 2.07, and the manual is now laser-printed. The latest revision was necessitated by a problem when using the program with the Atari ST QL Emulator; operation on this combination is now correct. The program will be reviewed shortly. *MS-QLink* is at version 1.6 and has a new manual, also laser-printed. The new version provides two pages of Help when F1 or F3 are pressed.

Hearing about computing in another country can be interesting. A guest in our house at the time of writing is a young man from East Germany, who is a computer enthusiast. From what one reads in the newspapers, it would be reasonable to expect him to have an ancient Spectrum, which cost a lot of money, and be using primitive programs, written by himself. Not so. He has a fast PC (25 MHz), with two hard disk drives, and plenty of up-to-the-minute software.

Bryan Davies considers 'a new phase in the QL's life'.

The only thing that is what I had expected is that almost all the software is pirated. He has been comparing prices here with those at home, and tells me everything is much more expensive here. So far, the only thing (of any sort) that he has found cheaper in the UK is a postage stamp! Maybe this situation will change, when the effects of the initial, artificial exchange rate between East and West German currencies wears off. A local QL friend demonstrated *Qpac* to our guest (who uses Microsoft *Windows* on his PC), and he was duly impressed that a computer which he had never before heard of had something similar to what he used; he had not been aware that there was a multi-tasking 8-bit computer.

This brought up the inevitable discussion on text only — as opposed to graphics — interfaces, and I found myself in the minority (as usual). There is a lot in common between *Qpac* and *Windows*; both give a very attractive screen presentation, and require a mouse, to be used sensibly. On the surface, both programs allow the user to do almost anything, simply by playing tunes on mouse buttons. Coincidentally, I had just been *trying* to get two *Windows* demonstration programs working, so was ready for some discussion (to put it politely) on the subject of text-versus-graphics interfaces.

Space required

Although there are similarities between the QL and PC products, there are some big differences, the most obvious one at first being the space required to house what are, after all, only utilities. Would you believe that the *demonstration* *Windows* program requires about 5 MB of hard disk space? Since most of the program functions in the demonstration version are disabled, one could expect the 'real thing' to need a couple of other computers networked-in to get it running. Actually, it's not that bad, but you need about 4-plus MB of disk space for the program alone

(without demonstration files and functions I would not use). At this point, you can start looking for space to house the programs you actually work with. It's a bit like saying that you, as a Quill user, need 50-100 KB to do your work, but another 500 KB is needed to connect you to Quill.

We are at the start of what may be a new phase of the QL's life, with the advent of the *Miracle Gold Card*, and the 2MB of memory it provides for us. For us, that 2 MB will allow the running of several, big programs, as well as something like *Qpac*. The poor PC user, trying to do a similar thing, wouldn't even get the utility program working reasonably, let alone have any applications programs working as well. No, I'm not confusing disk storage with memory. *Windows* also requires large amounts of memory to work properly. My German friend has a 'swap file' of 8 MB set up, so that *Windows* can handle the switching from one program to another, as well as its housekeeping functions. Even so, one of my own reasons for not using *Qpac* is that it uses more memory than I can spare — on an 896 KB system. It is to be hoped that any programmer developing utilities for the QL in the future does not follow the PC route, now that there is 2 MB available, and write utilities that take up more space than the programs we want to use.

That's one point of view. The other main one appears to be that the utility program is the most important thing, and is what computing is all about. That is, a computer should not be for doing work on, but for playing with. Fortunately, there's enough choice of software for the QL to suit both types of user fairly well. It does seem to me though that we are short of some good utility functions, regardless of which user-category we fit into. None of the housekeeping programs I have seen are sufficiently convenient to use, or flexible enough, to meet my needs. It was a pleasant surprise to find that the *WinBack* hard disk backup program is easy to understand and use, and does the job quite well, without forcing the user to jump through a hoop by involving him/her in a complex interface. No need for a mouse, no masses of keypresses, no massive memory demands, a simple and clear book of instructions; what could be wrong with it? It does have some limitations, but they are not that great, and later versions should overcome them. The program writer's

SHOOTER

M S O L V E D

name was not previously known to me, which makes it even better that the program is worth having. It is desirable to have some new programmers on the scene, to fill the gaps left by those who have gone other ways in recent years.

You don't see much mention of the Z88 computer in these pages, which is rather a pity, as it is a very useful little computer, and it is one of Clive Sinclair's brain-children also. It is usual to see at least one stand of Z88 supporters at the All Formats computer fairs, and there are quite a few programmers beavering away to produce all the little goodies needed for the Z88. Being a portable machine, it needs links to other types of computer, and it is well provided for in this respect.

There is a good link program to the QL, and another to the PC. A good example of the usefulness of the Z88 was the job a QL using friend did recently. He wrote a report on (one of) his Z88 while on-location in Brazil, and then brought the report back here stored in the machine. We used a link program to transfer the text file to a PC, and a format-conversion routine (supplied with the conversion program) to put the text into suitable form for loading into the WordStar word processing program, on a PC. A disk was then sent back to Brazil for incorporation into a larger document, being prepared on a PC.

The same process can be used to transfer text files to QL *Quill*, *text87* or *Perfection*, although there is then no need for the rather odd conversions required with WordStar. The *Pipe Dream* text format from the Z88 can be used without conversion, but it is then necessary to make use of the Search and Replace facility within the receiving WP program to get rid of some unwanted characters.

An odd point to note from Kaga-Taxan users who have bought a laser printer is that it is preferable to use the FX-80 driver of *test87* rather than the Canon one, when printing on the laser in its FX-80 emulation mode. Some errors in alignment appear

in the samples included with a recent article appear to have been due to using the Canon driver, which has some functions which are not standard to the FX-80. The sample given here was done with the GQ-500 laser driver, to show how well proportional spacing and justification is handled.

In the March issue, **Edward Jones** enquired about adding a second hard disk drive to his Thor. He mentioned Rodime as a possible supplier, since the original Thor drives came from them, but that company does not seem a suitable one to contact for a single drive now, partly because of the continuing financial problems

Readers Letters

the company has but mainly because it deals with OEMs (original equipment manufacturers) and its units are not usually seen for sale to the general public.

The Thor used an SCSI interface, allowing several devices (of different types) to be chained together. You can buy SCSI drives from several suppliers who advertise in PC magazines, but there might be some risk of incompatibility, depending upon the degree of faithfulness with which CST/Rodime and current drive suppliers followed the SCSI specification. Another potential problem area is the use of termination packs on drives; presumably, the Thor requires packs on some drives in the chain, but not on others. SCSI drives tend to cost more than standard, MFM/RLL, drives.

We advise users to buy products with credit cards because there is a reasonable chance of getting a refund from the card company if ordered goods do not turn up or are defective. You do need to make a claim against the card company within a fairly short period of time; don't leave it six months or more. What prompts the reminder now is another complaint about non-delivery of a keyboard, ordered from **Keyboard Products**. The delay for

David McCullagh had been seven months at the time he wrote to us (March this year). He is one of several people to make similar complaints. This supplier has so far failed to respond to requests for information on four other orders (but two of the purchasers have reported finally receiving keyboards).

Michael Jackson is one of the users who finally got a working keyboard from Keyboard Products. He reports that the first one received appeared to be 'a complete dud', but the current one works well. There have been some problems, such as unexplained system crashes. The crash problem may have been solved by the use of an anti-static mat under the new keyboard. The interface which has to be installed inside the QL is apparently sufficiently large to cause conflict with other add-ons, such as Minerva. The alternative keyboard interface, supplied by **Jürgen Falkenberg** (reviewed by Mike Lloyd a few months ago, and available from EEC Ltd) leaves more space for other interfaces to be fitted. Falkenberg is the supplier of the scanner which attaches to a printer; this also was reviewed favourably by Mike Lloyd, and is available from TK Computerware.

Anthony Magnus from the Balearic Islands is yet another reader to report difficulty obtaining a PC keyboard and interface from **Keyboard Products**. His order was placed in mid-September last year, he sent a reminder in early February this year, and he was still waiting when he wrote another in late March.

Jackson brought up the subject of the KBL 128 computer box, also supplied by Keyboard Products. Such a box is a good idea, but this particular one is too small to accommodate a typical collection of system devices, such as the QL plus a couple of floppy drives, maybe a hard disk drive, disk/memory interface. No doubt larger boxes can be obtained, but not in the form of kits made up specifically for the QL user. If you go to Quanta group meetings, you will see all manner of assemblies of QL's into boxes, some being quite neat and others rather messy, but the average user most likely has neither time, ability, nor inclination to do a fully DIY job like this. There is a small market area here for an enterprising one-man company to make some money.

Returning the David McCullagh (who must keep a Thor running permanently for

This text was written using *text87* and printed with the driver for the Epson GQ-5000 laser printer. It is left- and right-justified, and the printer font is the scaleable Epson Times, in 12-point size. The range of sizes available is 6-72 points. At small point-sizes, it may be preferable to use the Epson Helvetica, which is sans-serif -- no "tops and tails" on the character-ends -- and is also scaleable. The lack of embellishments on the characters improves readability, although the impression given may not be as "professional" as with Times. The caption below is in 12-point Helvetica Bold Underlined. You can also use the other resident fixed-height fonts -- Prestige, Courier, Modern and EDP.

LASER PRINT SAMPLE FROM TEXT87

writing letters to QL World), he has been contacted by **Jacques Flury**, who was kind enough to supply him with both a monitor cable and a set of Argos 6.34 ROM chips for the Thor XVI, and is going to supply connection details for using the Miracle printer interface with the Thor. McCullagh has been advised not to expect any response from Thor International. To be able to add a hard disk to the Thor XVI, special chips are said to be needed; it might be worth having these chips manufactured and the Swiss supplier **COWO** (see address below) has suggested that it could do the job if they had about 20 customers for their 0.5-2.0MB ram expansion units. They can also supply different versions of Argos, at about SFr. 100 each.

Back to the Serial 8056 printer again; **R Thompson** asks if anyone knows where to obtain continuous paper for this printer. We have been this route several times before, but I don't currently know of a supplier, so can anyone else offer names? Thompson is also in need of a manual for the printer; can someone supply a copy? The sample printout he sent had characters missing in several places, and some strange non-standard characters had been inserted at odd points. Missing characters are often a function of poor electrical connections between computer and printer, and the same fault can cause spurious characters.

Some very presentable prints of graphs (see illustration) were sent in by **Peter Hamill**. They were printed by a Tandy CGP-115 printer/plotter, from *Easel* data which had been saved to a ram file and then converted into a form suitable for the CGP. The routine for conversion was provided in the Quanta newsletter in November 1986, by J Stevens. The printer was quite cheap, but is out-of-production now; it might still be obtainable secondhand. If any other user needs help with graphs from this model, we can put him/her in touch with Hamill. The latter also mentioned an indexing routine for articles, documents etc, which is available (to Quanta members) from the Quanta library under the name *Indexer* (by Hugh Miall).

Reading a reader's letter reminded me

how contradictory are the impressions purchasers get of suppliers. Several people have said to me (and I've said myself) that **Miracle Systems** do not like answering letters; on the other hand, more people have given me instances of how helpful and generous Miracle have been in dealing with problems, going so far as to replace goods free-of-charge, and despatch goods without first requesting payment, in situations where there was no reason for the customer to expect to be treated so well. Longevity can generally be taken as a sign of reasonable customer service, in this business, and it is not by accident that **Miracle**, **Digital Precision** and **TK Computerware** have been with us for most or all of the QL's lifespan. There have been quite a few letters that I have written to suppliers over the past four years that have not received any answer, but I cannot recollect any instance where either D P or T K did not respond, and quickly.

The copy for the June issue had only just been sent off when **T K Computerware** called to say they had been contacted by **Mario Peer Bien** at the same time as he wrote to *QL World*, and that was the first they knew of his not receiving his cartridges. It is a bit late now (several months later) to start chasing the postal authorities about the package (which was despatched on 4th January last), but they will do so anyway. If the goods you order don't arrive, let the suppliers know about it within a reasonable time. Don't go to the extreme, though; one supplier called to say more than one customer had called at Easter time to complain about non-delivery of software *ordered during the previous day or two*, apparently quite oblivious of the fact that the postal service does not carry on as normal all through holiday periods (also, some people in business do like to take the odd day's holiday at such times). T K also commented that **K Blundell's** request for a replacement for his copy of *QSpell* was not being delayed by them, but had to be passed through other hands as well as theirs, since the program has not been sold for several years. It is only when a small batch of similar requests has accumulated that T K can get supplies of such out-of-

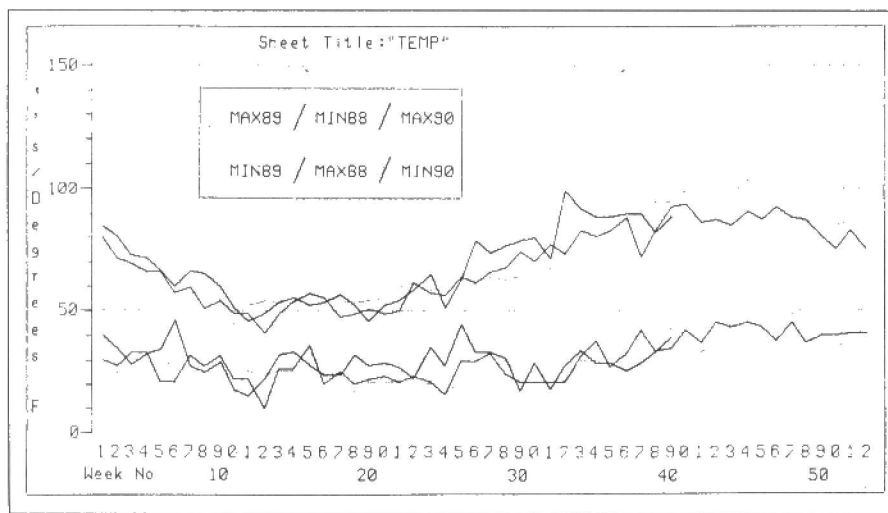
date items.

Hopefully, the change to a higher VAT rate will have passed without us having received a shoal of letters of complaint. Adverts placed before the budget — when suppliers had no idea the VAT rate was about to change — will have appeared with prices based on the old rate. Suppliers can't afford to pay the extra 21/2% themselves, on items supplied after the new rate came into effect, and they will be charging customers at the new rate. Bear in mind that advertisers (and magazines) can be held only to *their* advertised prices, not to the government's previous tax rate.

Jens Wildgruber wrote from Germany, about his experiences with Minerva V1.82. He is generally pleased with it, but has noted some incompatibility problems, for which he offered some 'work-arounds'. The SDUMP command on the Trump Card is disabled, unless another job is active in the background, and he suggests (for users of QPac) setting SYSMON or ALARM working. As noted previously, D P's *Turbo* won't start running with the usual CHARGE command, and you need to use EX PARSE_TASK (or equivalent) to start it off. *Ultraprint* (also from DP) crashes when used with its original extensions, but is ok if *Turbo Toolkit* is used instead. Turbo-compiled tasks work in 1-screen mode, but not in 2-screen mode, unless a change of address is made in the compiled task; he suggests replacing the address \$28010 by (ver\$(-2)+16), starting at byte 178 and 278. Hope that means more to you than it does to me! Wildgruber ends his letter by expressing the hope that QView will now stop developing *Minerva* and concentrate on making it work with other QL software.

CR Stephenson has received a full refund for a QL returned to **EEC Ltd.** and says he accepts that the delay in getting his money back was entirely accidental. I don't recollect seeing any other complaint about this supplier, so this incident would seem likely to be a 'one-off'.

We never go long without receiving a letter about printing problems, and the latest one comes from **P Land**. He has a Citizen 124D, and gets double-spaced output from both Quill and Abacus, but single-spaced from programs obtained from the book *The Working Sinclair* (by David Lawrence). The obvious reason would seem to be that he has failed to use the Design screen in the Psion programs to set the Spacing to 0, although the default setting is 0 and he (or someone else) would have previously had to change the setting to 1 to give output such as in his letter. If you don't read the User Guide, or do some tests, the menu options can be confusing, and I can see someone inserting the wrong number for the right reason, so to speak. Another possibility is that his *Printer_Data* file(s) contains the command to give a Line Feed with every Carriage Return, and his printer may automatically do that anyway, with the result that there are two line feeds for every carriage return.



All Formats

The next All Formats Computer Fair will take place at the New Horticultural Hall, Westminster, London on Saturday 22 June from 10am to 5pm. Tickets are £3. A new recorded information line about the fairs is open on 0898 299 389.

THOR BULLETIN

Thor enthusiast Malcolm Smith has sent the first copy of *Mjolnir* – the Unofficial Thor Users' Magazine, from Norway, in his own words 'a simple, honest eight-page affair' containing letters in response to his original appeals for interest, some tips, and many queries. The bulletin is very simply produced by photocopier, but clearly printed and laid out (with the exception of one page which perhaps should have been retyped to save space).

Mjolnir looks to QL World like a simple and honest attempt to start a Thor self-help group which, from the comments we receive, is sorely needed. Malcolm tentatively suggests a starting subscription of £5 for four issues, and the pilot issue available for 40p. Norway, he points out, is more expensive than the rest of Europe, and we can confirm that it is not the ideal place to put down in at 11 o'clock at night with electrical problems and no hotel booking.

Contact Malcolm at Statsrad Ihlensvei 66B, N-2010 Strommen, Norway. No doubt he would appreciate an International Reply Coupon with general enquiries.

A DUTCH TREAT

Dutch user group Sin QL Air report that the first Continental Pure QL Microfair in Eindhoven, Holland on May 6th went well.

CHM Biemans, Editor of the bulletin Quasar, writes: "Miracle introduced the Gold Card in the Netherlands, and Digital Precision's Freddy Vaccha held his inaugural Perfection speech, during the first Continental Pure Microfair in Eindhoven. In St. Joriscollege School in Eindhoven there were a considerable quantity of QL users from all over Europe, hoping to get the first release of DP's new wordprocessor Perfection. In the beginning it seemed like a speech about the whole of DP's list except Perfection, but after some hints that his time was up, finally the new WP was demonstrated and it looked so good that several users bought the test version straight away, to get the complete program delivered after some weeks.

"This type of lecture/demonstration was so popular that it will be continued at future meetings, where one or two producers of interesting programs will be invited.

"Cowo from Switzerland demonstrated the newly-renamed program QTop, with many utilities and multitasking features. This program also looked good and many visitors bought one after a clear audio-visual presentation.

"After the teach-in, there was a lecture from the Belgian Progs, with the recently developed DataDesign, which updated the previous QL World test of the program.

"The final lecture came from our own members about their QL-Unique system. It contains hotkey and multitasking configurations so that each user can devise a personal configuration to suit his or her needs. The club's promoter on this subject is Laurens Kremer, Magelerweg 53, 7683 VM Den Ham, Netherlands.

"The most surprising feature of the fair is the introduction and demonstration of the new Gold Card from Miracle Systems. This provides a QL68000 2MB with a speed of 12.16MHz, full 16-bit width, to fit into the QL's expansion slot instead of the Trump Card. It includes a battery-backed real time clock/calendar, and the disk drives

can handle 1.44MB or 3.2MB disks.

"Phil Borman of Quanta had been asked to bring copies of *The Definitive Handbook* by Jan Jones, and told us that about 20 had been sold. The charming team from EEC were there, and we are only sorry that they had only a few Keyboard-90 Interfaces available. Jurgen Falkenburg himself could not be there. We also missed Lear with their PCB-CAD program, and don't understand why there is not more publicity about this nice item, which we are sure many users would be interested in.

"Laurence Reeves of QView, together with TF Services, also had too few of the newest Minerva V1.89 roms, but they promised us that soon this new version will be ready for mailing, with the newest manual.

"We had a very good first continental fair in Holland."

Sin QL Air's foreign contact person is Marco Holmer, J P Coenstr. 61, 3531 EN Utrecht, Netherlands. The Membership Secretary is Bob Visser, Snelrewaard 6, 2904 SN Capelle, a/d IJssel, Netherlands.

Archive Desktop Bug Probe

Robin Stevenson reports: "Several readers have had problems with the to-do list program from the Archive Desktop series, and the cause seems to be a bug in Archive. If Archive files are reset or selected, when the file has only just been created, under cer-

tain (as yet unidentified) circumstances, Archive goes haywire, and starts filling the drive with blank records.

To avoid this situation in the to-do list, the program should be modified so that there is never an empty file. Add the following lines to the end of

the 'TODOFILE' procedure. An 'end of file' record will be inserted when you create a new list file:

```
let DATE$ = '2099/12/31'
let DAYS = days (DATE$)
let TEXT$ = 'END OF FILE'
append
```

"My apologies to readers who have struggled with this. If anyone can shed more light on the bug, perhaps they could write to *Open Channel* with details."

THE NEW USER GUIDE

In the fifth part of our New User Guide, Mike Lloyd looks at data statements in Superbasic, and the usual data input sources; keyboard, microdrive and floppy disks.

SECTION FIVE

So far in this series all the information given to the computer has come from SuperBasic commands such as 'LET x = 4'. In real life, programs have to cope with data entered by users.

SuperBasic is extremely rich in keywords associated with obtaining data from a variety of sources. There are three main sources of data input available to Basic programmers: the keyboard, microdrive and floppy disk files, and from special data statements within SuperBasic programs themselves. The latter will be dealt with on another occasion.

The keyboard provides the most direct contact with the user, and often provides the most problems. Users are notoriously prone to typing in the wrong thing, so the only way to trust them is not to trust them at all.

Data files on microdrives or disks and data statements within SuperBasic programs can be relied on to contain valid information provided that the programmer has full control over what is placed in them. If they contain information written by users then it is usually best to treat their contents with caution.

Input to a SuperBasic program can be obtained using the keyword INPUT. INPUT is very closely related to the PRINT statement with which you are now familiar. At its simplest, INPUT is followed by a single variable name:

INPUT

INPUT number

If the above line is typed in as a direct command, the cursor will flash on the next free line of the default window. Anything typed at the keyboard will appear in that window until the Enter key is pressed, at which point the computer will attempt to make the variable 'number' equal to whatever has been typed in. Try it out, and prove that the input has been accepted by typing the following command:

PRINT number

With luck, the output from the PRINT command will be identical to the input from the INPUT command. However, if what you typed was not a valid number the computer will respond with an error message, and if you typed a very large or a very small number the computer will have displayed the result in scientific notation. Sticking to numbers with less than six digits will avoid the distraction of scientific notation for the moment.

INPUT's similarity to PRINT shows in the way that it can be preceded by an AT statement to determine where on the screen the input line will appear; it is affected by PAPER and INK statements; a channel number can be included in INPUT statements; and any number of inputs can be obtained provided that each variable name is separated by a valid print separator. It is therefore possible to write code such as:

```
AT#2; 4, 7
PAPER#2, 5
INPUT#2; A, B$, C
```

Basic's fondness of halting processing with error messages is one of its biggest weaknesses, and nowhere is this more irritating than in the way in which Basic objects to invalid data entered in response to INPUT commands. Fortunately, INPUT statements can also accept text data simply by using string variables, such as:

INPUT text\$

No matter what is typed into a text variable the QL will never complain that the input is invalid, so it is a good rule always to use string variables with INPUT statements. If necessary, numbers can be converted, or more properly 'coerced', from text format to the computer's internal numeric format. This is demonstrated by the following program fragment:


```
INPUT text$
LET square = text$ * 2
```

COERCION

For coercion to work properly the string must begin with a valid number, for example it might read "123 High Street", otherwise we are back to an error message.

The simplest way to obtain trouble-free numeric input has been published in *Sinclair QL World* previously. The trick is to add a leading zero to all strings before coercing them. **Listing one** shows exactly how this can be done by rewriting last month's multiplication table program so that it accepts input from the user before printing a table.

LISTING 1 - MODIFIED MULTIPLICATION TABLE

```
100 PRINT "MULTIPLICATION TABLE"
110 INPUT "Enter number "; INFO$
120 LET NUMBER = "0" & INFO$
130 IF NUMBER = 0 THEN STOP
140 CLS
150 LET COUNT = 1
160 PRINT NUMBER; " x "; COUNT; " = ";
170 PRINT NUMBER * COUNT
180 IF COUNT = 12 THEN GOTO 100
190 LET COUNT = COUNT+1
200 GOTO 160
```

As you begin to develop programs requiring user input, the following sequence of statements will feature regularly in your programs:

```
PRINT "Enter your name"
INPUT name$
```

Computer language developers have added an extra facility to the INPUT command to assist in this circumstance. You can put text strings into an INPUT statement which will be printed out prior to the cursor appearing for input. The two lines above can be encapsulated in a single INPUT command such as:

```
INPUT "Enter your name" \ name$
```

You should by now be well acquainted with the association between the keyboard and input on the one hand and screen windows and output on the other. Communication between the computer and these devices is by means of 'channels', which can be imagined as conduits for Ascii characters. The concept of channels is central to the way the Qdos and SuperBasic make communications easy within any QL system.






When the QL is switched on three channels are opened by default, one for each of the screen windows. As a special arrangement for typing commands the keyboard is initially associated with the #0 window, also known as the command window.

When an INPUT statement is used, the keyboard is temporarily attached to the #1 window, aka the default window. If the INPUT statement includes the #2 channel identifier, the input appears in the #2, or listing, window. As we will soon learn, channels can also be opened to a printer and to files on microdrives and diskettes.

To open other windows new channels are designated with the OPEN command. OPEN must be followed by a channel number and by a device identifier. The identifier can be extended if necessary to include all sorts of extra information which will where possible be avoided in order to keep things simple.

The five device identifiers commonly used on QLs are listed in **Figure one**, although QDOS recognises a sixth, NET, which is used to connect QLs together in a network. Third party manufacturers have extended QDOS to recognise devices not originally included in the basic QL. The best examples are the now ubiquitous FLP1_ and FLP2_ device names for floppy disks.

QL DEVICES AND DEVICE IDENTIFIERS

	scr_ (INPUT not allowed)
	con_ (INPUT allowed)
	ser1_ & ser2_
	flp1_ & flp2_
	mdv1_ & mdv2_

There are two types of screen window. Windows opened as consoles allow the INPUT command to be used in association with them, but windows opened as screens do not permit INPUT to be used. All three default windows are consoles.

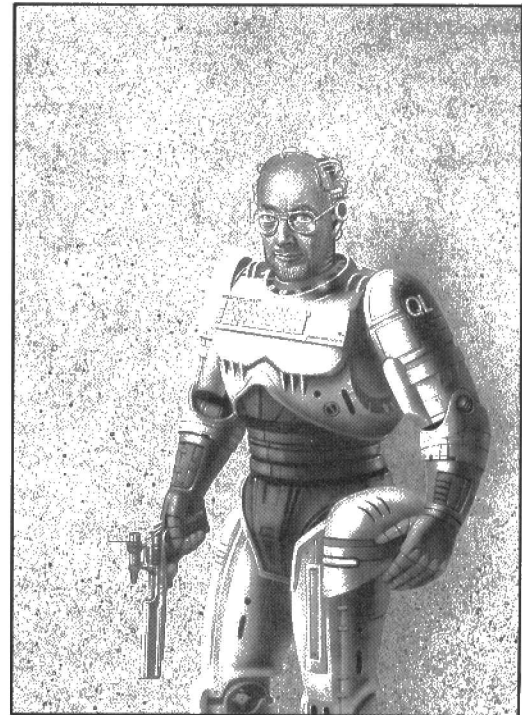
OPEN

QL World / QLAW
The QL in the

1990s!

Giant Joint

READER QUESTIONNAIRE



QLAW (the QL Advancement Working Group) in co-operation with *Sinclair QL World*, has drawn up a detailed questionnaire with the aim of producing a comprehensive profile of the QL-using community in the 1990s.

QLAW is a non profit making interest group founded by Public Health Physician, medical statistician and QL user Dr. Sohail Bhatti. The group's long-term aim is to discover what QL users want from their machines, to co-ordinate skill, experience and talent within the QL community, to provide a pool of expertise to beta-test new developments on request, and ultimately to work towards the development of a 'Super QL' with the co-operation and knowledge of major developers.

As a first step towards this goal, *QL World* is publishing QLAW's questionnaire, along with a series of questions designed to find out more about what QL World readers want from their magazine.

The questionnaires will be analysed and the results maintained on a database by QLAW. Names and addresses are requested. QLAW stresses that confidentiality will be maintained; only official representatives will have access to the information in the questionnaires. If you do not wish to include your name and/or address, however, your replies will still be of interest.

We are particularly keen to know as fully as possible the details of your machines and how you use them, so please try to answer every question. Sinclair roms can be identified by typing PRINT VER# and reading back the rom (JS, JM etc.) from the screen. Minerva owners will already know their version numbers.

We suggest that you look through the questionnaire before filling it in, as some items may appear in greater depth in later questions.

This questionnaire is also appearing in a similar form in the monthly *Quanta* bulletin, and in CGH Services' QL Technical Review.

QL users wishing to join QLAW or learn more about the group's stated aims will find more information in this month's *QL Scene*.

A typical response to published questionnaires is around two per cent of readers canvassed. For a few quiet moments and the price of a stamp, you could be helping to put the QL ahead of the crowd in the 1990s. Please return your questionnaire to us at QLAW!

There is no fixed time deadline, but Dr. Bhatti hopes to produce an analysis by August 1991. This questionnaire can be pulled out of your QL World, leaving the rest of the issue intact.

QLAW has no association with QL World, Maxwell Specialist Magazines or MCPC Ltd.

Please read the notes!

Notes: 1. A Levels-pre-higher education. If studying for a Degree, tick box for appropriate establishment & put course under "other" 2. Give the title of the job as well, using the "other/job" space. If retired then tick RETIRED box and also the box which most closely represents the major occupation of your life 6. Sinclair Rom name is obtained by typing PRINT VERS. Fill appropriate box and for each QL put the rom on dotted lines 7. Gold card owners should fill in "Disk+Memory Interface", giving 2Mb as a capacity. Trump card drive capacity is number of drives supported. Give as many details as you can about any machines or peripherals you use e.g. RAM, hard disk, make, etc. 13. Whether the option exists or not give your opinion on your most desirable one 18. The Quill refers to the program from Gilsoft NOT the word-processor. Use the current exchange rate to calculate £ equivalents. Filling in this questionnaire and returning it implies consent to contents being placed on a computer database.

QUESTIONNAIRE

All names and addresses remain confidential and will not be released without consent. Please read the notes at the beginning before answering the questions.

Personal details: ☐ Male ☐ Female Age

1. Education: ☐ ☐ ☐ ☐ ☐ ☐ ☐ *for Scotland H Grade = A level*
(HIGHEST ATTAINED) O Levels A Levels College Polytech University Other.....

2. Occupation: Teaching ☐ Managerial ☐ Divinity ☐ other/what? ☐
(Fill in Professional ☐ Scientific/Tech ☐ Student ☐
two if retired) Skilled Manual ☐ Computing ☐ Medical ☐
Retired ☐ Manual ☐ Arts ☐

3. Gross Income: ☐ ☐ ☐ ☐ ☐ ☐
(1K=1,000) <£5000 £5-10K £10-20K £20-30K £30K+ Not telling

4. Marital Status: Married ☐ Single ☐ Widowed ☐ Divorced ☐

5. Years with QL: ☐ ☐ ☐ ☐ ☐ ☐ ☐
<1 1-2 2-3 3-4 4-5 5-6 6+

6. How many working QLs do you have? ☐ ☐ ☐ ☐ ☐
(Put roms beside box for each QL if known) 0 1 2 3 4+

7. Which of the following peripherals do you have?
How many?

Want it? Yes!

Internal memory	<input type="checkbox"/>	How many K	Manufacturer	<input type="checkbox"/>
External memory	<input type="checkbox"/>	How many K	Manufacturer	<input type="checkbox"/>
Printer	<input type="checkbox"/>	Which model (main one)		<input type="checkbox"/>
Disk interface	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Trump Card	<input type="checkbox"/>	How many K	Drive capacity?	<input type="checkbox"/>
Disk + memory interface	<input type="checkbox"/>	How many K	Make + model	<input type="checkbox"/>
Hard disk interface	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Keyboard	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
System box/cabinet	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Disk drives [3.5in]	<input type="checkbox"/>	Capacity	K Manufacturer/model	<input type="checkbox"/>
[5.25in]	<input type="checkbox"/>			
Monitor	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Digitiser/Scanner	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Eprom programmer	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Modem	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Centronics interface	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Mouse	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
CAPS lock led	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>
Backplane	<input type="checkbox"/>	Manufacturer		<input type="checkbox"/>

Others (please specify)

8. How many other local QL users do you know? ☐ 0 ☐ 1-5 ☐ 6-15 ☐ 16+

9. Do you have another QL-specific machine? No: ☐
If so what make and model? ☐ Amiga ☐ AtariST ☐ Thor

(How much ram etc.)

10. Do you use any other computer? No: ☐ Yes:

PC(IBM compat)	<input type="checkbox"/>	Dragon	<input type="checkbox"/>	Spectrum	<input type="checkbox"/>	CPC	<input type="checkbox"/>
ST	<input type="checkbox"/>	Tandy	<input type="checkbox"/>	C64	<input type="checkbox"/>	MSX	<input type="checkbox"/>
Amiga	<input type="checkbox"/>	PCW	<input type="checkbox"/>	MSX	<input type="checkbox"/>	Console (any)	<input type="checkbox"/>
BBC	<input type="checkbox"/>	Archimedes	<input type="checkbox"/>	Macintosh	<input type="checkbox"/>	Other (what?)	<input type="checkbox"/>

Specify details (disks etc)

11. What is your use of the QL? (0=nil, 1=rarely, 2=sometimes, 3=regularly, 4=frequently, 5=always)

Word-processing	Basic Language
Machine Code	Computer Languages
Databases	Graphic Design
Educational	Games
Adventures	Program Development
Desk Top Publishing	Business
Hardware Design	Other (What?)

12. How many hours did you spend with your QL in the last typical week?

<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Hrs with non-QL
<1	1-2	2-4	4-7	8-14	15-28	29+

13. Which of the following, would you most wish? (0=no, 1=perhaps, 2=useful, 3=great, 4=love it)

Any disk drive	1.4MB Disk drive	20MB Disk drive	Plug in PC
Hard disk drive	512K Memory expansion	Gold Card	Use PC peripherals
Better keyboard	Better display resolution	Better sound	QL on a PC
Better printer	Cabinet/case	Monitor	Mouse

14. What is your main interest in the QL? (fill two only)

Word processing ☐ SuperBasic ☐ Machine Code ☐ Languages ☐ Databases ☐ Graphics ☐ Educational ☐ Games ☐
 Adventures ☐ Prog Develpt ☐ DTP ☐ Business ☐ Hardware ☐ Simulations ☐ Spreadsheets ☐ Other

15. Which is the machine you most want to buy?

Archimedes 3000 <input type="checkbox"/>	386 PC <input type="checkbox"/>	"SuperQL" <input type="checkbox"/>	AtariTT <input type="checkbox"/>	MegaSTE <input type="checkbox"/>
Macintosh Classic <input type="checkbox"/>	Thor <input type="checkbox"/>	Amiga 3000 <input type="checkbox"/>	Amiga 1500 <input type="checkbox"/>	Other <input type="checkbox"/>

16. How much money (total) have you spent on buying software for the QL?

<input type="checkbox"/> <£100	<input type="checkbox"/> £100-500	<input type="checkbox"/> £500-1000	<input type="checkbox"/> £1-2000	<input type="checkbox"/> £2K+
--------------------------------	-----------------------------------	------------------------------------	----------------------------------	-------------------------------

17. Have you received a QLAW questionnaire from another source? yes/no where?

18. How well do you know?: 0=ignorant 1=passing acquaintance 2=understand 3=proficient 4=expert

Perfection	Text87	Editor	Spy
QD	C	Forth	Pascal
Fortran	SuperBasic	BCPL	CPM
QLiberator	Turbo	Archive	Abacus
Easel	QPAC	Toolkit2	Flashback
DataDesign	ICE	PageDesigner2	ProPublisher
Assembler	Conqueror	Solution	QMON
GraphQL	TechniQL	3DPrec	EyeQ
Painter	GRAM	ACT	*The Quill
QDOS	MSDOS	Hardwre Design	Cash Trader

* See notes

QUESTIONNAIRE

19. Which word processor (if any) do you most often use?

.....

20. For QL information, do you read:

QL World Quanta QL Technical Review Other

How frequently?

Since when?

How satisfied?

(Rate 1 for poor to 10 for good)

21. How do you obtain your copy of Sinclair QL World?

Newsagents shelf ☐ Newsagent order ☐ Magazine subscription ☐ Borrowed ☐

Do you often see QL World on your newsagents shelves? Yes/No

22. Do you ever experience difficulty getting QL World from:

Your usual source? Yes/No Any alternative sources? Yes/No Through you subscription Yes/No

Comments (if any):

23. Why do you read QL World? Please indicate importance of factor:

	Very	Fairly	A bit	Not
Right price				
Supplier advertising				
News and new product information				
Hardware reviews				
Software reviews				
Tips on using commercial programs				
SuperBasic programming				
Machine code programming				
Problem solving (any kind)				
General interest				

24. In general, do you find QL World:

Well balanced

Dwells too much on particular areas (which?)

25. Is there any kind of article you would like to see more frequently in QL World?

.....

26. In general, do you find articles in QL World:

Too simple ☐ Too technical ☐ Just right ☐

27. On a scale of 1 (poor) to 10 (good), how do you rate:

The overall presentation of QL World?

The overall value of QL World?

28. Would you be interested in joining or supporting QLAW? Yes ☐ No ☐

For more information on QLAW, see QL Scene, page 19.

NAME + ADDRESS

.....

.....

..... POST CODE

POST THIS QUESTIONNAIRE TO QLAW, 4 WASDALE AVE, BLACKBURN, ENGLAND, BB1 1XD.

© QLAW 1991 QW.

A printer can be attached to the QL via one of the two serial port sockets at the back of the computer. Serial ports are so called because each of the bits of a byte of information are passed one after the other down a single wire, in serial fashion. The alternative arrangement is to have the bits of each byte travel simultaneously down eight wires, ie in parallel.

All QLs have two microdrives units fitted to their superstructure to allow files to be saved and accessed. When a channel is opened to a microdrive it is connected to a named file on a microdrive cartridge loaded onto the specified microdrive.

These days, many QL systems include one or two floppy disk units. Floppy disk drives are much faster and more reliable than microdrives, and floppy disks hold about seven times more information than a microdrive cartridge. Floppy disks are also extremely cheap, working out at around 75p per megabyte (roughly 1,000 kilobytes or a million bytes) whereas £20 worth of microdrives are needed to hold the same amount. Of course, before switching to floppy disks the cost of the disk drive must be considered, but the QL becomes a much more useful, responsive, reliable and fast machine with a disk drive attached.

If a new screen window was required, one which needed to accept INPUT commands, it could be opened with the following command:

```
OPEN#3, con_
```

The window will have default values for its size and colour which can be changed with appropriate commands, such as:

```
WINDOW#3, 140, 20, 10, 10  
PAPER#3, 2  
INK#3, 0
```

There is no requirement to open channels with consecutive numbers, although to do so is usually a good idea. It is also advisable to re-use channels, simply by opening them as a new device. Novices often fall into the trap of unnecessarily opening large numbers of windows at various screen locations. A more economical approach is to have just one window channel which can be relocated to suit what is being printed at the time.

In order to use a printer in SuperBasic a command such as:

```
OPEN#5, ser1_
```

must be used.

More often than not the device name needs extending to include essential parameters so that the computer and the printer are synchronised. Additionally, the speed of data transmission might need to be changed using the BAUD command, but these refinements will be discussed in a later part of the *New User Guide*.

Once a printer channel is open, the PRINT command can be used to send data to the printer, for instance:

```
INPUT "Type your name.", name$  
PRINT#5, "Your name is "; name$
```

Some commands associated with screen channels, such as PAPER and INK, have no effect on the printer AT and CLS and all of the graphics commands are also prevented from reaching the printer. A useful substitute for AT which can be used with printers and screen windows is TO, which moves the print position to a specified character position on the current line. In SuperBasic TO is embedded in a PRINT statement, as in:

```
PRINT#5, "This is the"; TO 20; "20th tab stop"
```

TO cannot be used to move the print position to the left of its current location or to the right of the window or page print area.

Programs can be listed to the printer by adding the appropriate channel number, such as:

```
LIST#5
```

Sometimes the last line of text sent to the printer will not be printed because it is held in a memory location called a buffer until some more text comes along to shunt it to the printer. It is possible to force the last few characters out of the buffer by closing the channel to the printer. Unsurprisingly, the keyword is CLOSE and the format for the command is:

```
CLOSE#5
```

By the way, it is a good idea to get in the habit of always using the same channel number for the printer.

Information is stored on microdrive cartridges and floppy disks in files. A file can contain a SuperBasic program, commercial software such as Psion Quill, data such as a word-processed file or some spreadsheet figures, or a representation of a screen display.

TO

CLOSE

To open a new file using SuperBasic, a command from the OPEN family is followed by the name of the device and the file itself, eg:

```
OPEN_NEW#4, mdv 1_demofile
```

Once the file has been opened it can be written to using the PRINT command in exactly the same way as PRINT has been used for screen windows and the printer.

```
PRINT#4, "Here is some stored text"  
PRINT#4, " - and a second line of text"
```

As was the case with the printer, such incidentals as colour information and the AT command cannot be used with files.

Once you have finished writing to the file it can be closed with the CLOSE command:

```
CLOSE#4
```

It is extremely important to remember to close all channels linked to files before switching off the QL or removing the microdrive cartridge or floppy disk from its drive. If a file is open when the QL is turned off its contents will almost certainly be destroyed, and other files on the same medium may no longer be accessible. If you accidentally remove a file medium without closing channels connected to you can usually avoid disaster by re-inserting the correct medium. However, there is no substitute for rigorously following this advice:

CAVEAT!

ALWAYS CLOSE FILE CHANNELS

File contents are accessed by connecting them to a channel with one of two keywords from the OPEN family: OPEN or OPEN_IN. OPEN allows both PRINT and INPUT commands to work, but OPEN_IN only allows INPUT commands to be used, thus protecting the contents of your file from being overwritten by the accidental use of the PRINT command.

It is good practice, at least for the time being, to use OPEN_IN rather than OPEN to read back information from files. You would only use OPEN to write to files which already exist and which contain data you no longer want to keep. To retrieve the two lines of text sent to the file above and print them on the screen, the following commands are required:

```
OPEN_IN#6, mdv 1_demofile  
INPUT#6, text$  
PRINT text$  
INPUT#6, text$  
PRINT text$  
CLOSE#6
```

Points to note are the choice of OPEN_IN as the safest way of accessing a file for read-only operations, the use of INPUT to read back each line from the file and the CLOSE statement at the end of the sequence. the program as it stands is fine for a two-line file, but for longer files some sort of loop needs to be developed containing the INPUT and PRINT statements.

While it is possible to save a program listing like this:

```
OPEN_NEW#3, flp 1_program  
LIST#3  
CLOSE#3
```

it is rarely a good idea.

SuperBasic includes a single command called SAVE which does all of this. The equivalent SAVE command to the sequence above is simply:

```
SAVE flp_1 program
```

To retrieve the program into memory again, type:

```
LOAD flp1_program
```

Once the program has been loaded it can be LISTed, EDITed and RUN. Alternatively, if all you want to do is run the program, type:

```
LRUN flp 1_program
```

LRUN is shorthand for 'LOAD and RUN'. SuperBasic programs can be 'daisychained' so that each program loads its successor by including a LRUN command in the program. However, it is a rule that only one SuperBasic program can be in the QL's memory at one time, so both LRUN and LOAD have the effect of wiping out any program already in memory.

SAVE and LOAD

LRUN

Be Reconciled

I Nigel Bates makes personal finance into an Abacus tutor.

The value and usefulness of the QL and of the software available for it is tempered by the degree of simplicity with which that software available can be made to run smoothly. While there are undoubtedly several examples of good software to help keep track of one's personal finances and even those of small businesses, many people would be satisfied with a simple but effective application which did not require hours studying the manual or elaborate setting-up procedures.

The beauty of the Psion programs and the testament to their continued usefulness is the ease with which data can be entered into them and useful information obtained immediately. While a very substantial number of articles have appeared describing various uses for Archive, its close relative Abacus has tended to receive less attention until recently.

Reconciliation

The following is a simple one-screen application to enable reconciliation of bank statements with cheque-book records. There will be those that have got such a tight control of their personal finances that they will have invested in a much more comprehensive piece of software to do this and other things for them. But, for quite a number of people, some simple, straightforward method of carrying out this regularly-occurring procedure would be very welcome.

Where do you begin if you decide to attempt a reconciliation? What do you make of all those standing order payments, cash machine receipts, cheque stubs, bank charges – or bank interest if you're the thrifty type – and countless other details of the average current account? At first

sight, it's not obvious and, unless you are an accountant or bookkeeper, you probably may not be able to devise your own system to reconcile your version of things with the bank's.

Some years ago, I had an account with one of the smaller banks which provided statements on the back of which was a detailed set of instructions on how to carry out a reconciliation. I'm surprised that all banks do not provide such a useful service. I decided to adapt this to the QL. I choose to use Abacus rather than program in SuperBasic as it is possible to achieve very useful results with a spreadsheet in a fraction of the time it takes to program. And when you want to sort out your finances, the last thing you want is to get tied up in a loop!

The following Abacus application was one of the very first I developed back in the heady days of 1984 when the first QLs were rolling off the production line. I am still using it to reconcile my cheque book with my bank statement and I have found it absolutely invaluable. With the addition of extra memory, disc drives, Minerva and QPAC2, my six-and-a-half year old QL is now ten times the machine it was then, and beats the socks off most PCs!

A reconciliation is basically designed to match every item in your records with a corresponding entry in the bank's record, ie the statement. In actual fact, it is not in principle necessary to do any arithmetic at all, merely to tick off the items as they appear. In practice, however, because of the possibility of confusion between several similar items such as cash-dispenser withdrawals, it is better to keep a running total of the actual amount of all transactions. Also, because of the time it takes for cheques etc, to be cleared by the bank, the figure the bank regards as your

actual balance may never exactly match that in your records. Therefore, it is desirable to choose a definite point of time and attempt to match the cheque-book record with the bank statement by taking account of all other items which have been omitted in one and included in the other and vice versa.

This application was designed for someone who prefers to keep details of all bank account transactions on cheque-book stubs. Of course, it would be possible to use Abacus itself to keep a list of these details or even to modify the application described below to do so, but this would entail a necessary degree of complexity which would compromise the principle of ease of use. In practice, as long as we continue to use cheque-books, this seems to be the most logical place to maintain a current record. If the day ever comes that all transactions are carried out electronically, then the application could be modified to take account of this.

Full records

To be able to use the application you will need to start getting into the habit of keeping full records of all the financial transactions on your bank account by collecting all those cash-dispenser slips which, previously, you may have thrown in the bin. Unless you are the type of person who can remember every penny you have spent and fifteen birthdays in your head at the same time, make a list of Standing Order or Direct Debit mandates on your bank account on a sheet of paper. Use your previous statements to help fill in the gaps in your memory. Keep this sheet for reference purposes and update it as soon as changes occur in the details. Make sure you always write up all cheque stubs fully as you make them out. Later, write into your cheque stubs at the correct places details of all other withdrawals and deposits made by you between the dates of the cheques. I find it useful to use a sharp pencil and to write the details in code form, eg 12/10 -20.00 CD (£20 withdrawn by cash-dispenser on 12th October) or DD (Direct Debit) as necessary or make up your own. Even on a cheque stub, it is amazing how much information you can fit.

Obviously, at the start, you will not have a previous reconciled balance with which

```
F3 : C[+0]R[-3]+C[+0]R[-2]
F12 : askn(C[-1]R[+0]+ " ")
F17 : askn(C[-1]R[+0]+ " ")
F19 : -askn(C[-1]R[+0]+ " ")
F20 : C[+0]R[-3]+C[+0]R[-2]
F25 : C[+0]R[-3]+C[+0]R[-2]
F35 : rept("-",width())
F37 : askt("Enter month ")
F38 : rept("-",width())
F39 : rept("-",width())
F40 : sum(C[+0]R[-10]:C[+0]R[-2])
F41 : IF(abs($E19-$E10)>=0.005,"NOT RECONCILED","RECONCILED")
F52 : -$B17
F56 : C[+0]R[-3]+C[+0]R[-2]
```

to begin. You will have to accept the bank's version of events and use the final balance of the most recent statement you have received. Write this figure in, again in pencil, somewhere near the top of the cheque-stub whose date follows the date of this balance. Subtract from this balance the amounts of any withdrawals and add to it the amounts of any deposits whose details you have written in. Don't worry if you think you have not remembered everything. In any case, you will not know beforehand how much interest on your account you are due or, if you are not so lucky, how much in bank charges you have to pay. They need only to be taken account of in the month of the statement on which they appear; and that statement will give you those details.

Set-up

To set up the application, first press F2 to get full-screen display followed by F3 D A ENTER to turn off Abacus's auto-calculate on input. Set the default display of numeric data with the units command: F3 U D ENTER ENTER to give two decimal places. Change the column widths for each column with the grid command. Set column A to a width of 9 spaces by pressing F3 G W 9 ENTER A ENTER A ENTER. Change each of the widths of columns B to E in the same manner setting B to 8, C to 3, D to 30, and E to 8 spaces respectively. You may wish to blank out amounts in the spreadsheet which are zero by pressing F3 D B ENTER.

Enter the following formulae into the appropriate cells as given. Where text items are being entered, you need only type the first quote ("): Abacus fills in the other for you.

```
A2 "a/C No.:"
A4 "DEBITS OUTSTANDING"
A5 "Cheque No."
A16 rept("-",width0)
B2 "(Your own account no.)"
B5 "Amount"
B16 rept("-",width0)
B17 sum(B6:B15)
B18 rept("=",width0)
C1 "CHEQUE-BOOK AND
STATEMENT RECONCILIATION"
D2 askt("Enter month")
D4 "Cheque-book balance"
D5 "Statement credits not in recs."
D7 "Subtotal"
D8 "Less stmt. debits not in recs."
D9 rept("-",width0)
D10 "Adjusted cheque-book balance"
D11 rept("=",width0)
D13 "Final statement balance"
D14 "Add total lgmts. not on stmt."
D16 "Subtotal"
D17 "Less 'DEBITS OUTSTANDING'"
D18 rept("-",width0)
D19 "Account balance"
D20 rept("=",width0)
E3 "£"
E4 askn(D4+"")
```

```
E5 askn(D5+"")
E6 rept("-",width0)
E7 E4+E5
E8 -askn(D8+"")
E9 rept("-",width0)
E10 E7+E8
E11 rept("-",width0)
E13 askn(D13+"")
```

	A	B	C	D	E	F
1:						
2:				F37		
3:						
4:					F12	
5:					F12	
6:					F39	
7:					F20	
8:					F19	
9:				F39	F39	
10:					F25	
11:				F35	F35	
12:						
13:					F17	
14:					F17	
15:					F39	
16:	F38	F38			F56	
17:		F40			F52	
18:	F35	F35	F39		F39	
19:					F3	F41
20:				F35	F35	

```
E14 ask(D14+"")
E15 rept("-",width0)
E16 E13+E14
E17 -SB17
E18 rept("-",width0)
E19 E16+E17
E20 rept("-",width0)
E19 if(abs(SE19-SE10)>=0.005, "NOT
RECONCILED", "RECONCILED")
```

To improve the appearance of the application, the text in cell B5 should be right-justified (F3 J ENTER ENTER R ENTER) and that in cells A17 and E3 should be centred (F3 J ENTER ENTER C ENTER).

Automatic resize

The rept0 function is used to separate different parts of the spreadsheet by drawing lines of dashes and double dashes. The inclusion of the width0 functions in the formula means that the lines are resized automatically if the column-widths are changed. Two extreme useful functions in Abacus are askt0 and askn0. These effectively make an Abacus spreadsheet behave as if it were a SuperBasic program by generating prompted requests for data entry. Both may take text parameter or, as here, a reference to a cell containing text. When the spreadsheet is executed, Abacus will work through these functions presenting them as queries in the prompt line at the bottom of the screen with any text prompts they contain. In the case of askt0, you may press ENTER if there is nothing you wish to type in. With askn0, however, which expects numeric input, if there is no number to enter you must type zero (0) followed by ENTER otherwise Abacus will refuse to continue.

Save this as a file called CHEQBOOK_aba

and use it as a blank template when starting out to do the reconciliation. You may have more than one account on which you wish to use it.

When the day of reckoning arrives and you receive your bank statement, don't be tempted to put it aside to deal with it late. If there are any errors on the statement –

and believe me errors do occur – you will want to inform your bank as soon as possible. In my own experience, I find the banks are getting better at correcting their own errors before they get through to you; and if there is a mistake on the statement it is not unusual to see a correcting payment further down the sheet.

Statement dates

Note the date of the statement. You may, if you wish, carry out the reconciliation for any period you like; but I find it simpler to use the dates given at the beginning and end of the statement even if these do not correspond to a calendar month. Go through the cheque-book stubs for the period since the last statement and before the latest date of the present statement and tick off each debit which also appears on the bank statement. Any left unticked should have their amounts as numeric data. You can expand the space for this if you have a lot of debits outstanding, but you will need to move the formula in cell B17 further down and change it to cover the greatest range of cells; and you will also need to alter the formula in cell E17 which uses an absolute cell reference. Change it to refer to whatever cell you have moved the formula originally in B17 to.

Press F3 X to start eXecution. The program prompts you for the necessary information it needs. Make sure you enter all debts as positive amounts – the program will put in the signs for you. However, if either the starting cheque-book balance or the final statement balance are negative, ie overdrawn, then enter these with a minus sign in front of them.

Abacus will ask you to type in the month in question. This is merely for record

purposes. The heart of the program is in two parts, the first of which basically corrects the deficiencies in your record keeping by including those unexpected items that have appeared on your bank statement, and the second part corrects the deficiencies of the bank statement by adding on deposits that you have made which have yet to be cleared before they appear in the bank's records.

Estimate

The program asks you for the balance you have estimated in your cheque-book up to the date of the statement. This is the figure you will have written in pencil at the bottom of the latest cheque-stub before the statement date and corrected for any later transactions of other types that may have occurred. The program then asks for the total amount of any credits that have appeared on your statement that you have not taken account of in your cheque-book record. This could include bank interest, money paid into your account from other accounts, etc. Next, you are asked for the

records but which have not appeared on the statement. Enter this value in response to the prompt. Lodgements you know of made by other people into your account are probably best left out of the reckoning for the present. They will appear on the following month's statement and you will take account of them under the 'Statement credits not in records' query of cell E5.

Comparisons

Abacus then goes away and does its sums. If every conceivable item has been included correctly, the figures in cells E10 and E19 will be identical and cell F19 will show the word RECONCILED, otherwise it will say NOT RECONCILED. This is an interesting example of the use of the if0 function in Abacus. The program compares the difference in the values of the contents of the two cells to see if it is greater than or equal to £0.005, ie a halfpenny. If it had been asked to check that they were exactly equal, then it would almost always be NOT RECONCILED as Abacus calculates its numbers to sixteen places of decimals

that it was not included in the cheque-book balance entry in cell E4. If the number calculated in cells E10 and E19 are different, the difference between them may well reveal what item has been omitted from your records, especially if it is an odd amount such as a cheque. If there are several items, it may not be so easy; but in any case, if you continue keeping records, you will be able to achieve a full reconciliation on your next statement. A most unfortunate situation to find yourself in would be if the bank had actually themselves made an error on the statement you first tried to reconcile! After much head-scratching and frayed nerves, you might eventually become convinced of the usefulness of the program.

Cash withdrawals

In fairness to my present bank, I have to say that it has always been me that has been wrong and not my bank statement. Perhaps others have not been so well served! It usually comes down to a missed cash withdrawal such as the time the cash dispenser ran out of printed receipts and you forgot to make a note of it in your portable organiser! Having carefully kept all such receipts, I can truthfully say that, with a little checking, I have never failed in over six years to reconcile my own records with my bank statement.

No need to scroll

The program is designed in such a way as to make it unnecessary to scroll around the spreadsheet or to use the Abacus window facility. Once you have satisfactorily reconciled your records with the bank statement, you may save the information as an Abacus file in the usual way, and print it out if you wish. I use a file name such as CHBK0CT_aba to refer to my October cheque-book reconciliation. Once reconciled of course, when you come to use the program on your next statement, you can load the previous month's file and remove the details of those outstanding debits (use F3 R) that now appear on the latest statement and add in any new ones that have arisen. Then you can save the new information with a different name and delete the old file. In this way you need only have one Abacus file saved on your disk or microdrive at any time. If you do wish to keep all reconciliation data files, then use unique filenames such as CHBK0190_aba for your January 1990 file, for example.

In all the years I have been using this program, the feeling of satisfaction brought about by the appearance of that heartwarming word 'RECONCILED' on the screen at the end of my labours has been immeasurable and has given me great confidence in the many dealings I have had with my bank. I trust it may also be of service to you.

A B C D E F			
CHEQUE-BOOK AND STATEMENT RECONCILIATION			
A/C No.: 12345678 OCTOBER '90			
f			
4 DEBITS OUTSTANDING	Cheque-book balance	212.79	
5 Cheque No. Amount	Statement credits not in recs.	6.79	
7 000123 21.75	Subtotal	219.58	
8 000125 20.40	Less stat. debits not in recs.	-8.00	
9 CD 8 Oct 10.00	Adjusted cheque-book balance	211.58	
10 000126 33.83			
11 CD 16 Oct 20.00			
12			
13	Final statement balance	292.39	
14	Add total lgmts. not on stat.	25.17	
15			
16	Subtotal	317.56	
17 Total 105.98	Less 'DEBITS OUTSTANDING'	-105.98	
18			
19	Account balance	211.58	RECONCILED
20			
CELL A1 EDIT USE A1:F20 MEMORY 453K			
CONTENTS EMPTY			

value of any debits which appear on your statement but which also you have not taken account of in your cheque-book records. This will include bank charges, card fees, and any standing order or direct debit payments which you have forgotten to include in your own records or any cheques written before the date of the first reconciliation which have only now appeared on your statement.

The program then asks you for the final balance from your bank statement. There is one remaining item to complete the picture. It is the total value of any cash/cheques that you have lodged and taken account of in your own cheque-book

and would only by chance ever get them exactly equal.

First attempt

At this stage, it is quite possible that the program is telling you that things have not been reconciled! When testing this application out for this article, by pretending that I had not been keeping a full set of records, I got very close on the first attempt. The problem turned out to be a cheque which had been written some weeks before the date of the account balance that I had decided to start with. This meant

DIY TOOLKIT



Simon Goodwin digs out more hidden Qdos extensions and reveals the most versatile DIY keywords yet.

This month's *DIY Toolkit* project introduces five new SuperBasic keywords: MTRAP, QTRAP, BTRAP, ADDREG and DATAREG. These provide direct access to the hidden features of Qdos used to implement tasks and SuperBasic commands.

Qdos is the collection of machine-code routines that give the QL its unique character. The kernel of Qdos is built into the computer, but programs and devices can add more.

Qdos facilities range from simple things like character input and output to hard-disk maintenance calls undocumented in Qdos books or Sinclair's Technical Guide.

Most of the facilities of Qdos are controlled by TRAPs — concise 68008 machine-code instructions that switch the processor into 'supervisor mode' and divert execution to system software.

TRAP #1 calls 'manager' routines that deal with the allocation of tasks, devices and memory. TRAP #2 can open and close channels, format drives and delete files. Once a channel is open, operations use TRAP #3, the 'I/O utilisation' trap.

More than a hundred management and channel use routines are built into every QL or Thor rom, designed to be called from machine-code using TRAP instructions. Most books on Qdos list the traps and their parameters; I use Andy Pennell's *QDOS Companion* and Adrian Dickens' *QL Advanced User Guide*. Similar information has appeared in other books, articles and the *QL Technical Manual*.

There is not room to list all the standard routines here, although I shall give examples of some that are not normally accessible from SuperBasic. The full set is tabulated, with examples, in the new *DIY Toolkit*, Volume T for Traps.

I have planned a set of 'general' Qdos access routines for years, but deferred the task while waiting for details of new Traps and considering the development

of further specialised routines. I was impressed by the Thor's IO_TRAP keyword, and decided to go ahead with a *DIY Toolkit* set after comments from Lisbon's Tiago Freitas Leal in letters to *QL World* and *Quanta*.

Timeouts

Tiago argued for standard Basic TRAP routines, pointing out that some features of Qdos, like timeouts, are not supported by existing Toolkits. "If these Toolkits did include extensions for all possible Qdos traps, they would grow beyond the eeprom size, or take too much ram." My generalised Trap keywords need less than 400 bytes.

The new commands are general-purpose, and rather arcane by *DIY Toolkit* standards. The five extensions can handle a total of 21 parameters between them — but sensible defaults are set so you can get by with something as simple as MTRAP 0, to read the Qdos configuration, or QTRAP #0, 21 to move the cursor up a line.

My routines concentrate on TRAP #1 and #3 calls. They are hidden, numerous and useful, while other rom routines are already accessible to SuperBasic programmers. The standard CALL allows the use of vectors. TRAP #2 routines are well served by CLOSE, DELETE, FORMAT and many variants of OPEN, including the pipe commands QLINK and CONNECT.

Qdos is an extendable system, so add-on programs, roms and devices can add extra TRAP facilities. For instance, Sinclair roms lack facilities to rename or truncate files; these were added by Tony

Table: Qdos late arrivals:

TRAP #1

MT. TRA	\$24	Set translate tables	(JS/MG)
MT. ???	\$25	Sinclair assigned, unknown	(JS/MG)
MT. LTHNG	\$26	Link a Thing in Ram	(SMS-2)
MT. RTHNG	\$27	Remove a Thing	(SMS-2)
MT. UTHNG	\$28	Use a Thing	(SMS-2)
MT. FTHNG	\$29	Free a Thing	(SMS-2)
MT. ZTHNG	\$2A	Zap a Thing	(SMS-2)
MT. NTHNG	\$2B	Next Thing	(SMS-2)
MT. STHNG	\$2C	Next Thing user	(SMS-2)

TRAP #3

SD. ROP	\$37	Screen raster operation	(Sinclair)
SD. DOT	\$38	Set points, pixel coords	(Sinclair)
SD. LINEP	\$39	Draw lines, pixel coords	(Sinclair)
SD. TOPW	\$3A	Bring window to top	(THOR)
FS. RENAME	\$4A	Rename file, A1 -> New name	(TK2/QFLP)
FS. RENAME	\$4B	Truncate file	(TK2/QFLP)
FS. DATE	\$4C	Set/Read file dates	(CST/Miracle)
FS. MKDIR	\$4D	Make file into directory	(THOR)
FS. WATER	\$4E	Read Thor 'watermark'	(THOR)
FS. VERS	\$4E	Set/Read file version	(Miracle)
FS. XINF	\$4F	Read Drive details	(Rebel/Miracle)

Listing 1

```

* QL WORLD DIY TOOLKIT - Direct Qdos TRAP access routines
* Version 0.9, Copyright 1991 Simon N Goodwin
*
* initialise lea.l define,a1 A1 -> extension details
      movea.w #110\w,a2 Fetch BP.INIT word vector
      jmp (a2) Add these extensions

*
* MTRAP TRAP1_KEY [ ,D1 [,JobNum] [ , D2 , D3 , A0 & A3 , A1 ]
*
* mtrap capa.l a3,a5 At least one parameter?
      beq.s bad_param If not, moan at once
      moveq #112,d5 Separator bit mask: 01110000
      and.b l(a3,a6.l),d5 Read the first separator
      subi.b #48,d5 Binary "\n" pattern: 00110000

*
* D5 is Zero if the (assumed) first parameter separator is "\n"
*
      movea.w #118\w,a2 Fetch CA.GTLIN vector
      jsr (a2) Get long integers
      bne.s bad_return Give up if fetch fails
      cap.w #6,d3 Too many parameters?
      bhi.s bad_param
      movem.l 0(a1,a6.l),d0/d1 Pick up a couple
      tst.b d5 Job Number parameter?
      bne.s get_regs

*
      move.l d1,d5 Save the Job Number in D5
      moveq #0,d0 MT.INF trap key
      trap #1 Find the system variables
      move.w d5,d1 Make ID from the Job Number
      swap d1 Prepare to receive the Tag
      lsl.w #2,d5 Scale offset for long words
      add.l 104(a0),d5 Add job table base offset
      cmp.l 108(a0),d5 Check total against the end
      bge.s off_job Offset beyond end of table?
      movea.l d5,a0 Dn/An is not so orthogonal
      move.l (a0),d5 Pick up the Job base address
      bml.s off_job That Job is not running!
      movea.l d5,a0 A0 points to the job header
      move.w 16(a0),d1 Copy the tag from JB.TAG
      swap d1 D1 = TAG:NUM = Qdos Task ID
      move.l 0(a1,a6.l),d0 Restore first parameter D0.L

*
* get_regs movem.l 8(a1,a6.l),d2/d3/a0/a1
      movea.l a0,a3 Help for MT.IPCOM
      trap #1
      lea.l regtable,a2 Store registers in table
      movem.l d0-d3/a0,(a2) N.B: A0 is saved by TRAP #1
      moveq #0,d0
      rts

*
* off_job moveq #-2,d0 Report "invalid Job"
      bra.s bad_return Route all errors one way
      not_open bra.s bad_return Report "channel not open"
      bad_param moveq #-15,d0 Report "bad parameter"
      bad_return rts

*
* BTRAP : QTRAP # CHANNELX , TRAP3_KEY [ , D1, D2, D3, A1, A2 ]
*
* btrap moveq #1,d7 Addresses are inside Basic
      bra.s trap3 Lau might start there: 'D7=0!'
      moveq #-1,d7 Indicate absolute addressing
      trap3
      movea.w #118\w,a2 Fetch CA.GTLIN vector
      jsr (a2) Get some long integers
      bne.s bad_return Give up if that fails
      subq.w #2,d3 Too few? Minimum is A0 , D0
      bml.s bad_param Channel & Trap Key needed
      move.w 2(a1,a6.l),d0 Get LOW word of parameter 1
      mulu #40,d0
      add.l 48(a6),d0 Add channel base offset
      cap.l 52(a6),d0 Check it is within the table
      bge.s not_open
      move.l 0(a6,d0,l),d0 Pick up the Qdos channel ID
      bml.s not_open Dops, the channel is closed!
      move.l d0,a0 Set A0 for the call

*
      subq.w #3,d3 D1, D2, D3 required?
      bpl.s get_set Yes - so read the lot
      movea.l 4(a1,a6.l),d0-d2
      moveq #-1,d3 Default timeout -1, forever
      bra.s relatively No need to set A1 or A2

get_set subq.w #2,d3 Too many parameters?
      bhi.s bad_param
      movem.l 4(a1,a6.l),d0-d3/a1/a2
      relatively tst.l d7 Is this a BTRAP?
      bml.s all_set No! QTRAP is ready
      trap #4 Warn Qdos of A6 offsets
      trap #3
      lea.l regtable,a2 Find register storage area
      movem.l d0-d3/a1,(a2) N.B: TRAP #3 saves A1
      moveq #0,d0
      rts

*
* Functions to read register values from the REGTABLE area
*
* addreg capa.l a3,a5 No parameters expected
      bne.s bad_param
      moveq #16,d5 Skip stored D0.L .. D3.L
      bra.s default Continue as for DATAREG

*
* datareg clr.w d5 Default = D0, report code
      capa.l a3,a5 No parameter?
      beq.s default
      move.w #112\w,a2
      jsr (a2)
      bne.s bad_return
      subq.w #1,d3 Just one parameter?
      bne.s bad_param
      move.w 0(a1,a6.l),d5 Permit access to D0 .. D3
      cap.w #3,d5
      bhi.s bad_param
      addq.l #2,$58(a6) Update BV.RIP stack pointer
      lsl.w #2,d5 D5 := D5 * 4 (LONG offset)
      default lea.l regtable,a2
      * move.l (a2,d5.w),d0 Read stored register value
      * Make D0.L into a 6 byte decimal on the RI stack
      *
      return_fp move.w d0,d4 D4.W will be exponent
      move.l d0,d5 D5.L will be mantissa
      beq.s normalised Zero is a trivial case
      move.w #2079,d4 First guess at exponent
      add.l d0,d0 Already normalised?
      bvs.s normalised
      subq.w #1,d4 No, halve exponent weight
      move.l d0,d5 Double mantissa to match
      moveq #16,d1 Try a 16 bit shift

*
* normalise move.l d5,d0 Take copy of mantissa
      asl.l d1,d0 Shift mantissa d1 places
      bvs.s too_far Overflow! must shift less
      sub.w d1,d4 Correct exponent for shift
      move.l d0,d5 New mantissa is more normal
      asr.w #1,d1 Halve shift distance
      bne.s normalise Try shift of 8, 4, 2 and 1
      moveq #6,d1 Six bytes please
      move.w #11A\w,a2 Check space with BV.CHRIX
      jsr (a2) No error return possible
      move.l #58(a6),a1 Get revised BV.RIP value
      subq.l #6,a1 Make room for one float
      move.l a1,$58(a6)
      move.l d5,2(a1,a6.l) Stack mantissa
      move.w d4,0(a1,a6.l) Stack exponent
      moveq #2,d4 Floating point result
      moveq #0,d0
      rts

*
* N.B: Register values D0-D3 & one address, total 20 bytes,
* re-use the DEFINE table RAM space. Move REGTABLE for ROM!
*
* regtable
* define dc.w 3 Three procedures
      dc.w qtrap-#
      dc.b 5,'QTRAP'
      dc.w btrap-#
      dc.b 5,'BTRAP'
      dc.w mtrap-#
      dc.b 5,'MTRAP'
      dc.w 0 End of procedures
      dc.w 2 Two functions
      dc.w datareg-#
      dc.b 7,'DATAREG'
      dc.w addreg-#
      dc.b 6,'ADDREG'
      dc.w 0 End of functions
      end

```

Tebby and others when the first disk systems appeared, and now work on micro-drives too, thanks to code added by *Super Toolkit 2* and *Minerva*.

QTRAP 74 lets you re-name a file on floppy and hard disk. A1 should point at the new name. QTRAP 75 truncates a file. *Minerva* and *Toolkit 2* extend the micro-drive handler to support these keys.

Current roms allow Trap keys between 0 and 127. My code passes a full long word so that it will work with future systems that may support thousands or millions of variations. Some of the less-well-known TRAPs are documented in the table.

Many readers will be interested in the new routines and trap numbers that have been added to control add-ons from *Miracle*, *Rebel* and *CST/Thor*. These can be

called with BTRAP or QTRAP, even if there is no new Basic command to use them, or you don't want to rely on the command that may be provided.

Hard disk

Several new TRAP codes implement hard disk operations that are not supported by standard Qdos. Some of these work with other devices, like Tony Tebby's revised ST drivers, but parameters and results vary between *Rebel*, *Miracle* and *CST Thor* implementations, despite Chas Dillon's best efforts to encourage standards. I shall explain the variations as they have been explained to me, and welcome information about changes or extensions.

The most useful new traps are the group with keys between 76 and 79. Like all TRAP #3 routines these work on channels, identified by an ID in A0, with extra parameters in data registers.

FS DATE lets you read or set the three date fields in the header of a file. D2 indicates the date concerned: 0 for update, 1 for last reference, and 2 for the backup date. If you only need to read the date, set D1 to -1; on return D1 will contain the date, in seconds since January 1st 1961, to suit QL DATE functions.

Dates have been documented, but not implemented, in QL file headers since the launch. The standard 'GetHEAD' trap FS. HEADR can read these dates, but FS. HEADS has never been able to set them.

To set the current date, call FS. DATE

Listing 2

```

100 REMark Sinclair QL World HEX LOADER v 3
110 REMark by Marcus Jeffery & Simon N Goodwin
120 :
130 CLS: RESTORE : READ space: start=ALCHP(space)
140 PRINT "Loading Hex..." : HEX_LOAD start
150 INPUT "Save to file...";f$
160 SBYTES f$,start,byte : STOP
170 :
180 :
190 :
200 DEFINE FunNotion DECIMAL(x)
210 RETURN CODE(h$(x))-48-7*(h$(x)>"9")
220 END DEFINE DECIMAL
230 :
240 DEFINE PROCEDURE HEX_LOAD(start)
250 byte = 0 : checksum = 0
300 REPEAT load_hex_digits
310 READ h$
320 IF h$="*" : EXIT load_hex_digits
330 IF LEN(h$) MOD 2
340 PRINT "Odd number of hex digits in: ";h$
350 STOP
360 END IF
370 FOR b = 1 TO LEN(h$) STEP 2
380 hb = DECIMAL(h$(b)) : lb = DECIMAL(h$(b+1))
390 IF hb<0 OR hb>15 OR lb<0 OR lb>15
400 PRINT "Illegal hex digit in: ";h$ : STOP
410 END IF
420 POKE start+byte,16*hb+lb
430 checksum = checksum + 16*hb + lb
440 byte = byte + 1
450 END FOR b
460 END REPEAT load_hex_digits
470 READ check
480 IF check <> checksum
490 PRINT "Checksum incorrect. Recheck data.":STOP
500 END IF
510 PRINT "Checksum correct, data entered at: ";start
520 END DEFINE HEX_LOAD
530 :
540 :
550 REMark Space requirements for the machine code
560 DATA 378
570 :
580 :
590 :
600 :
610 REMark Machine code data
620 DATA "43FA014434780110","4ED2B8CB67687A70"
630 DATA "CA33E80104050030","347801184E926658"
640 DATA "B67C000662504CF1","0003E8004A056828"
650 DATA "2A0170004E413205","4841E54DDAA80068"
660 DATA "BAA8006C6C282045","2A106B2220453228"
670 DATA "001048412031E800","4CF1030CE8082848"
680 DATA "4E4145FA00E248D2","010F70004E7570FE"
690 DATA "600670FA00270F1","4E757E0180027EFF"
700 DATA "347801184E9266F0","55436BEA3031E802"
710 DATA "C0FC0028D0A0030","B0AE00346CD42036"
720 DATA "08006BCE20405743","6A0A4CF10007E904"
730 DATA "76FF900A554382BE","4CF1060FE8044A87"
740 DATA "8B024E444E4345FA","007E48D2020F7000"
750 DATA "4E75B8CB6A07A10","80224245BCEB710"
760 DATA "347801124E926890","5343668A3A31E800"
770 DATA "BA7C0003628054AE","0058E54D45FA0048"
780 DATA "2032500038002A00","671C383C081FD080"
790 DATA "8B1453442A007210","2005E3A089049841"
800 DATA "2A00E24166F27206","3478011A4E92226E"
810 DATA "00865D892D480058","2385E8023384E800"
820 DATA "780270004E750003","FF38055154524150"
830 DATA "FF2A054254524150","FEB2054D54524150"
840 DATA "0000002FF760744","415441524547FF64"
850 DATA "0641444452454700","0000","*",32375

```

with D1 zeroed. Other values of D1 are treated as new dates to be set. This scheme suits the Rebel Hard Disk and early Miracle drives, but current Miracle models do not implement the reference date; the space is used to store a file version number instead.

IO. VERS takes and returns a parameter in D1. Zero simply reads the version number for the file identified by A0; -1 reads and increments the number. Other values set the version, between 1 and 65535.

IO. XINF reads full details of a hard drive into a 64 byte buffer pointed to by A1. After the call, with D1=0, the buffer contains the medium name, device name and number, followed by a non-zero flag byte if the drive is read-only. The next word indicates the number of bytes in each 'allocation unit'. It may range from 512 bytes to 8K or more, depending on the system.

The last three documented values are long words: the total number of allocation units available, the number free, and the length of the file header, usually 64 bytes. The standard FS. MDINF assumes a maximum of 65535 512 byte sectors, but IO. XINF allows drives to hold far more than 32 megabytes.

Graphic calls

The TRAP #3 MT. ROP is mentioned briefly in Sinclair's Technical Manual, along with some apparently unreleased pixel graphics calls, akin to DIY Toolkit's PLOT and DRAW. IT seems that these traps have never been implemented. Argos 6.41 rejects them, reporting 'bad parameter'.

Minerva 1.64 rejects SD. DOT and SD. LINEP, and crashes if I call SD. ROP with QTRAP #2,55 – the rom jumps to \$C558, which reads an absolute address from \$28494 (somewhere in Screen 1, or the Supervisor stack), and tries to jump there. This causes the crash, as the address

(\$2F0063B8) is not valid. This is a strange bug, but hardly serious.

MT. TRA is the equivalent of the TRA (TRANslate) command in later Sinclair roms. The next TRAP #1 key, 37, is reserved by Sinclair, but I do not know why. Other TRAP #1 codes, documented in the Quanta library, are provisionally assigned to Tony Tebby's long-running SMS-2 project. The obsolete QPTR Toolkit documentation listed a further 17 TRAP #3 routines, with codes between 108 and 127.

MTRAP takes between one and six parameters. The first parameter is mandatory and corresponds to the 'trap key' – the value of register D0 when the TRAP #1 is performed. This determines the operation performed by the trap, and is normally an integer between 0 and 35.

Optional

Five other parameters are optional, depending on the trap key and the operation to be performed. They are passed in data registers D1, D2, D3, and address registers A0 and A1.

Many MTRAP operations deal with Qdos tasks, normally identified by a 32 bit 'task ID' in register D1. The ID consists of two integer values packed into one long word, which makes it fiddly to use from Basic.

You have the option to provide just the task number, preceded by a backslash separator. In such a case, MTRAP converts the integer parameter into a task ID before passing it on to Qdos. For instance, MTRAP 9\1 release task 1. Task numbers are displayed by common toolkit commands like JOBS and LIST_TASKS.

I use MTRAP 9 to regain control of the system when I try to print having forgotten to connect the printer, or switch it on. At such times Qdos locks up the task that is trying to print, suspending it until the printer interface is ready.

If I turn on the printer or connect the cable at this late stage it is likely that some characters will be lost or corrupted as the connection is made. Just as it is best to turn off a hose before you re-connect it, I prefer to release the suspended task, giving a 'file I/O incomplete' message in *Quill*, and try again later when everything is set up properly.

Taskforce

This trick works with Devpac, Psion software, and other tasks that use Qdos to talk to devices. Psion users need *Taskforce*, or some similar multi-tasker, so they can issue the MTRAP command from Basic while the offending task is stuck.

MTRAP can be useful when setting or reading Minerva's extra MODE parameters. The DIY disk includes UNRESPR, MINMODE and DISPLAY%, which tells you which screen is being displayed.

QTRAP takes between two and seven parameters. First comes a SuperBasic channel number, followed by an operation key for TRAP #3. Standard roms accept keys 0 to 73, with a few gaps.

QTRAP #C%, 14 turns on the cursor in window C%, QTRAP #C%, 17, COL% moves it across to column COL%, while QTRAP #F%, 67, OFFSET winds the file pointer on channel F% back or forth by OFFSET bytes.

There are lots more examples – including JOBLIST, EDLIN, QDOS_VER\$, Minerva and heap routines – in DIY Toolkit Volume T, one of 16 volumes of programs, examples and text expanded from this series. The new volume assembler source for the TRAP keywords and details of all known Qdos calls.

Each volume costs £3, on our branded disks or your microdrive cartridges, plus a processing charge of £4 per order. Write to **DIY Toolkit at Cwm Gwen Hall, Pencader, Dyfed, Cymru SA39 9HA**, or call (0559)

Listing 3 Listing 2

```

100 REMark DIY TOOLKIT TRAP KEYWORD demonstration
110 REMark BTRAP, QTRAP, MTRAP, DATAREG, ADDRREG
120 REMark Copyright Simon N Goodwin 3 May 1991
130 REMark Version 0.5, uses DIY Toolkit BPEEKs
140 REMark Tested on Minerva 1.84 and Argos 8.41
150 :
160 DEFINE PROCEDURE JOBLIST
170 LET job=0
180 PRINT "SuperBasic version ";VER$
190 REPEAT show_em
200 MTRAP 2,job,0
210 job=DATAREG(1)
220 IF job=0 : EXIT show_em
230 PRINT "Task ",job," is at",ADDRREG!
240 PRINT " priority " (DATAREG(3) && 255
250 END REPEAT show_em
260 END DEFINE JOBLIST
270 :
280 DEFINE PROCEDURE SHOW_CURSOR(ch%)
290 QTRAP #ch%,14
300 END DEFINE SHOW_CURSOR
310 :
320 DEFINE PROCEDURE HIDE_CURSOR(ch%)
330 QTRAP #ch%,15
340 END DEFINE HIDE_CURSOR
350 :
360 DEFINE FUNCTION QDOS_VER$
370 LOCAL q$(4),v,w
380 MTRAP 0
390 LET v=DATAREG(2) : q$=""
400 REMark Unwrap LONG into STRING
410 REPEAT unwrap
420 w=INT(v/256)
430 q$=CHR$(v-w*256) & q$
440 IF w=0 : EXIT unwrap
450 v=w
460 END REPEAT unwrap
470 RETURN q$
480 END DEFINE QDOS_VER$
490 :
500 DEFINE FUNCTION SYSVARS
510 MTRAP 0 : RETURN ADDRREG
520 END DEFINE SYSVARS
530 :
540 DEFINE PROCEDURE EDLIN
550 LOCAL buf,limit%,time%,p,result
560 REMark Edits LENGTH bytes in SuperBasic's buffer
570 REMark (which is often used by resident procedures)
580 REMark - it's wiser to use your own memory, but this
590 REMark illustrates how BTRAP and BPEEKs can co-exist
600 REMark - and use a timeout, specified in seconds.
610 buf=BPEEK_L(0)+16
620 limit%=BPEEK_L(8)-buf
630 time%=10 : PRINT #0,time%:" seconds >";
640 REMark Waits for TIME% seconds, with 50 Hz displays
650 BTRAP #0,4,0,limit%,time%*50,buf
660 PRINT "EDLIN:" (DATAREG!
670 result=DATAREG
680 SELECT ON result
690 =0 : PRINT "No error "
700 =-1 : PRINT "Not complete "
710 =-5 : PRINT "Buffer full "
720 =REMAINDER : PRINT "Unexpected "
730 END SELECT
740 PRINT (ADDRREG-buf)"bytes used:"
750 :
760 REMark Echo buffer contents... if still there!
770 LET p=buf
780 REPEAT echo
790 PRINT CHR$(BPEEK(p));
800 p=p+1
810 IF p=ADDRREG : EXIT echo
820 END REPEAT echo
830 PRINT #0 : PRINT
840 END DEFINE EDLIN
850 :
860 DEFINE PROCEDURE SEEK(ch%,offset)
870 REMark Move file pointer by +/- OFFSET
880 QTRAP #ch%,87,offset
890 END DEFINE SEEK
900 :
910 DEFINE FUNCTION FILEPTR(ch%)
920 QTRAP #ch%,87,0 : RETURN DATAREG(1)
930 END DEFINE FILEPTR
940 :
950 DEFINE PROCEDURE TAB(ch%,col%)
960 QTRAP #ch%,17,col%
970 END DEFINE
980 :
990 DEFINE PROCEDURE FOUNTS(ch%,base1,base2)
1000 REMark Base =0 for the standard founts
1010 QTRAP #ch%,37,0,0,-1,base1,base2
1020 END DEFINE FOUNTS
1030 :
1040 DEFINE PROCEDURE RELEASE(job%)
1050 MTRAP 0\job%
1060 END DEFINE RELEASE
1070 :
1080 DEFINE FUNCTION JOBSPACE
1090 REMark Finds maximum free RAM for a task
1100 MTRAP 6 : RETURN DATAREG(1)
1110 END DEFINE JOBSPACE
1120 :
1130 DEFINE PROCEDURE HOLD_UP(job%)
1140 REMark Suspends the job indefinitely
1150 MTRAP 8\job%,0,-1
1160 END DEFINE HOLD_UP
1170 :
1180 REMark More in DIY TOOLKIT Volume 1 for Traps

```

384 574, 1pm-9pm.

Customised

Existing toolkit commands are customised for specific tasks, so they miss out some Qdos features in the interest of simplicity. Almost all channel traps allow a 'timeout', so that the program can continue, given an appropriate indication, if a device or channel remains 'busy' for a specified time.

Qdos lets you specify a timeout whenever you use a channel, but SuperBasic is not so generous. INKEY\$ is the only standard routine that lets you set a timeout. INKEY\$ (100) waits until a character is available, or two seconds (100 frames, at 50 frames per second) have passed.

It might be useful to use INPUT in a similar way, allowing the program to continue, using the text typed up to the deadline even if Enter was not pressed. This scheme is used in Di-Ren's excellent *Fleet Tactical Command*, written in machine-code. Now even Basic programmers can use timeouts in their programs.

The default timeout, -1, allows 'infinite' time. The Qdos scheduler continues to try to perform the operation until it works. Other possibilities range from 0 to 32767 frames — almost 11 minutes. Timeouts are minima; a timeout of zero means 'only do this if it can be done immediately.'

If a timeout of zero is used for display output, the action is performed at once

unless CTRL-F5 has been pressed to pause display output, in which case Qdos returns with DATAREG set to -1 (not complete). A result of zero, ERR. OK, indicates that the operation went smoothly. Thus you can write programs that do not get stuck or mess up the screen with trivia when it is paused with CTRL-F5.

BTRAP is a variant of QTRAP with special uses in SuperBasic programming. It lets you refer to the moving values inside the interpreter data area.

Buffer

For instance, SuperBasic uses a 'Buffer' to store temporary results, INPUT lines and data being copied. The Toolkit function BPEEK_L (0) finds the offset of the start, and BPEEK_L (8) points past the end of the buffer. The exact address depends on the value of register A6, which points at the start of the SuperBasic area, and may change at any time.

You can find things inside Basic with DIY BPEEKs, or Turbo Toolkit equivalents, but they are liable to move as Basic runs or other tasks use memory. This is no good if you need to pass an address to QTRAP.

The SuperBasic interpreter makes much use of Qdos, so it needs to be able to use buffers inside its movable space. DIY Toolkit's BTRAP extension uses the same mechanism to address memory inside

Basic.

BTRAP works like QTRAP but the address parameter is taken relative to A6, so that you can refer to the offset inside Basic, without worrying about where the interpreter may have moved its tables when the code is executed.

BTRAPS let you save and load data held inside the SuperBasic task, like array and string details. Be careful not to overwrite other things by accident.

Display founts are not expected to move, once set, so you should select them with QTRAP rather than BTRAP. QTRAP #c%, 37, 0, 0, -1, FOUNT0, FOUNT1 does the trick, assuming C% is the SuperBasic channel number and FOUNT0 and FOUNT1 are absolute addresses.

These keywords are great for experiments, as a single command will trigger any manager or channel use trap. The functions DATAREG and ADDRREG let you check the result, error code, and values of all returned parameters.

Flexible

The code is designed to be as flexible as possible. It will pass any value you supply — sensible or otherwise — so it should accommodate future Qdos embellishments as well as Sinclair's originals and current extensions.

The corollary is that the code can be dangerous in the wrong hands. 'Suck it and

see' programmers are at the mercy of the rom parameter checking. Inappropriate TRAPs may crash the system.

Avoid using or releasing memory that has not been allocated, and don't specify odd addresses or negative lengths for string text. Qdos assumes that such parameters are correct — as they should be, if they were correctly calculated, or returned by one of the other TRAPs.

It is often vital to check the error-code of TRAPs before assuming valid results. If an attempt to allocate memory has failed, you should not try to use the space. Often the error code indicates what a program should do next. If the cursor is already on the top line, QTRAP #0, 21 sets DATAREG to -4, signalling an 'out of range' error.

DATAREG is zero after each successful operation; otherwise it is a Qdos error code. Unless the value is zero, the operation failed and other results are not reliable.

Access

As far as I am aware, QTRAP, MTRAP and BTRAP give complete access to the parts of Qdos previously hidden from SuperBasic. The parameters and result functions have been chosen to let you set or read all the relevant registers.

In general, TRAP #3 calls pass parameters in registers D1 and D2, with the channel ID in A0 and timeout in D3. Address parameters are passed in A1 and A2. D0 is the error code, and other results are returned in D1 and A1. TRAP #1 calls are similar, passing data in D1, D2 and D3, but there is no channel ID to worry about, so addresses uses A0 and A1.

MTRAP copies its fifth parameter to A3 as well as A0, to suit MT. IPCOM, which uniquely takes a single address parameter in A3. Other calls take two address parameters in A0 and A1, so A1 is the optional sixth parameter of MTRAP. Save your work before you call MT. IPCOM for the first time — you can easily crash the 8049, or 68008, or both, if you get the command sequence wrong.

If several results need to be returned Qdos stores them in a buffer, addressed by A1. A1 should point at an appropriate memory space, starting at an even address. For instance QTRAP 0, 11, 0, 0, -1, BASE calls SD. CHENQ, which finds the window size and cursor position in characters, and stores four words of results at BASE. MTRAP 24 allocates common heap memory.

D1 is commonly used to return a single number. A few calls set D2, and MT. JINF returns D2 and D3, so DATAREG accepts an integer parameter between 0 and 3; the default is zero.

ADDREG takes no parameters, as Qdos calls only return a single address: A0, in the case of MTRAP, or A1 for QTRAP. BTRAP returns A1 relative to A6, to suit BPEEK and its ilk.

As usual, the keyword routines are listed in two forms. **Listing two** is a quick way to enter the code without using an assembler. The first part is the same every month. Use RESPR instead of ALCHP at line 150 if you do not own Toolkit 2. Only the DATA, from line 590 onwards, changes for each DIY project.

The loader reads equivalent machine code from DATA statements, and saves the code in a file. Once you've loaded that file, as follows, you can use QTRAP, MTRAP, BTRAP, DATAREG and ADDREG in your own programs:

```
base = RESPR (378)
LBYTES "file name", base
CALL base
```

Alternatively you can load the file directly with LRESPR or LINKUP. If you have already loaded or entered the new commands on a JM or AH machine before you loaded the code, you need to type NEW to make the extensions available.

Listing one is the corresponding assembly code program, written and assembled using HiSoft's DevPac. Type this text into your own assembler if you want to customise the code. Note that some assemblers expect word vectors like \$110\w to be entered as \$110.W, or simply £110.

The machine-code falls into three main chunks, wrapped around by initialisation code and data. The first chunk deals with MTRAP. It fetches long word parameters with CA. GTLIN, after checking the first parameter separator in case it is a backslash.

Bad values

If D5 is zero the second parameter is a job number. TRAP treats it as an offset in the Qdos task table, and uses it to find the full task ID. Bad values cause an 'invalid job' report.

The multiple move instruction MOVEM is the star of this listing. It is used to read several parameters from the maths stack at once, directly into data and address registers. It does no harm to pick up 'parameters' that have not been supplied. Code to count and check parameters would use up ram, complicate testing, and probably consume more processing time than it could save.

If explicit values are not supplied, registers are set to arbitrary values read from after the real parameters in memory. This does no harm, as the maths stack pointer BV. RIP is unmolested; in any case it is reset automatically when a resident procedure finishes. Unspecified registers may have any value. If the parameter value matters, supply it.

BTRAP and QTRAP share much the same code. The only difference is the TRAP #4 performed by BTRAP, immediately before the main TRAP #3. TRAP #4 sets bit 7 of the flag JB. RELA6 in the current task

header. The next TRAP #2 or TRAP #3 uses this as an indication that addresses are relative to A6, rather than absolute.

BTRAP and QTRAP report 'bad parameter' if you supply too few or too many parameters, and 'channel not open' if you get the first parameter wrong. Always check before using TRAP #4, or your code may abort leaving the flag set, causing bizarre effects when the next TRAP comes along.

The last chunk of code implements the ADDREG and DATAREG functions. Both return long integer results converted to floating-point form by the familiar RETURN_FP normaliser.

After each successful call the register values are stored in 20 bytes of memory are REGTABLE. The data labelled DEFINE is not used once the TRAP commands are linked to the system, so I re-use their space to hold the results. If you put the code in rom you will need to move REGTABLE to ram.

Redundant?

It might seem that DIY Toolkit is now redundant. Who needs further SuperBasic extensions, if everything is open to MTRAP, QTRAP and family?

The answer is that often special-purpose code is faster and friendlier than a sequence of TRAPs. Many programs can be written with a mix of Basic and TRAP calls, yet others — like my SET, DRAW and CLIP routines — add so much that is new that they are only practical in custom-crafted machine-code.

QTRAP and MTRAP are great when prototyping or testing new facilities and interactions, or when you want to write code that runs independently of proprietary toolkits. They shine when you need total control over errors and timeouts, or underlying routines, like FS. XINF, are not supported by commercial toolkits.

DIY Toolkit TRAP commands let you get straight at the latest versions of Qdos, without waiting for Toolkit writers to catch up.

I welcome your comments: applications for QTRAP and MTRAP that I may have missed, or ideas about extensions as yet unavailable to SuperBasic programmers. Please write to me, care of *QL World*.



SOFTWARE FILE

INFORMATION

Program: *Sector X*. 256K memory, Minerva-compatible.

Supplier: GCH Services, Cwm Gwen Hall, Pencader, Dyfed, Wales. SA39 9HA.

Price: £10.00 for 3 1/2 or 5 1/4 in disk (plus £1.00 p&p). £10.00 mdv or £12.00 two mdvs (inc. training game) plus 10% p&p.

SECTOR X

Just another horizontal scrolling shoot-em-up game?

Here are some of the best graphics I have seen for a long time on the QL. They must be a close rival to the public arcade machines. I am impressed.

I don't profess to be an expert on this type of game so I left the main testing to the neighbour's children. They found it addictive and unanimously sung its praises.

The scene is set in one Lunar 7, where evil hordes have taken over the five planets of the Sector X solar system. To restore peace on Sector X, you have a ship with limited fire-power, and must shoot down energy balls to get extra weapons. Lastly you have to destroy the mothership and the evil Command-Eye.

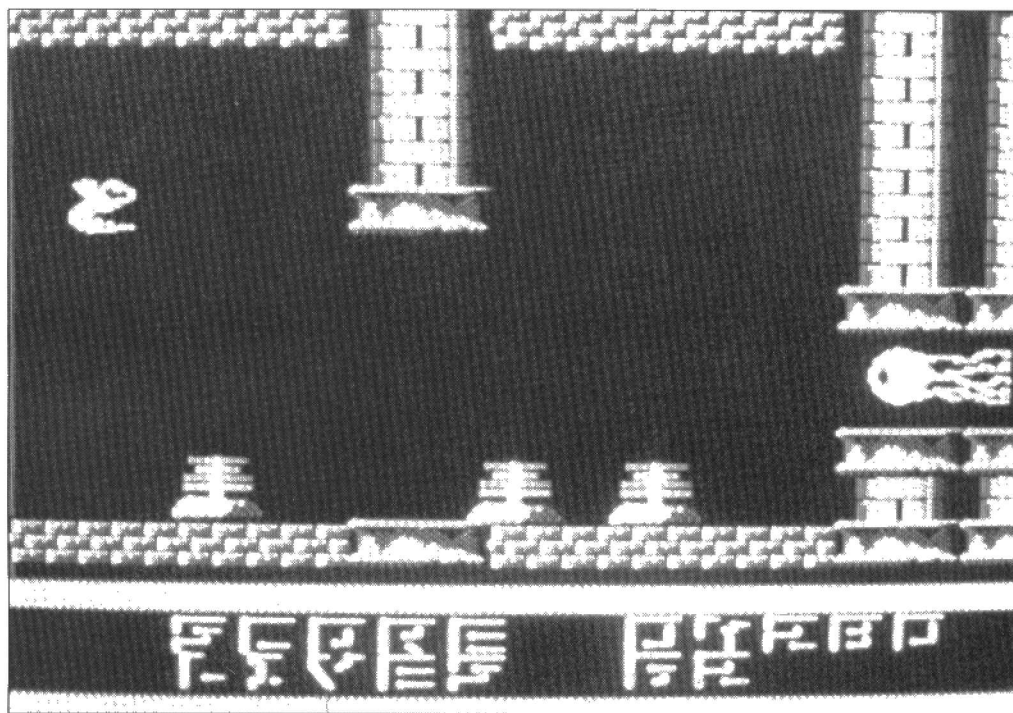
This is a fast action, reflex testing and difficult arcade game.

Training

Fortunately the author has provided a training mode which gives you auto-immunity against most of the aliens. This enables you to go through all the scenarios enjoying the superb graphics. I think my lasting memory will be having a discussion with my friends, well into the small hours, on the best way to get past the evil spring-loaded killer eyeballs which wang out at you with unbelievable ferocity.

You will do battle with

John Shaw (and his consultancy team) venture into orbit in search of the killer eyeballs.



seemingly invincible snowflakes, manic chess pieces and massive fleas. One valuable tactic I learned is the Harrier pilots' 'up and over' manoeuvre which seems to be the only way to kill creatures which appear behind you.

This is an arcade game, which should provide many, many hours of lasting pleasure; I recommend it without reservation.

QUANTA IN EAST ANGLIA

One area of the QL scene which has previously escaped investigation by many QL users is that of local computer clubs. Quanta, the QL User Group, has approximately twenty clubs or 'sub-groups', dotted in many places both in Britain and on the Continent, where Quanta members and non-members meet normally on a monthly basis. This is a report from the March meeting of a typical sub-group QMAS (Quanta Mid-Anglia), which holds its meetings on the second Monday evening of every month in a small church hall in Lolworth, a village just outside Cambridge.

The QMAS Club is fairly small, attracting each month about twenty-five members with varying skills, from those who have just mastered *Quill* to one member who designed parts of the QL.

Local groups dedicated to the QL are flourishing. Carl Watson reports.

ments in QL software and hardware being shown for the first time.

One piece of new software, the beta-test version of Digital Precision's *Perfection*, was being shown by one of the members at the Club, and it was clear that it is a very fast word processor indeed. Using what is known as a 'lazy screen' to speed up the cursor key movement, it also has full WYSIWYG (what you see is what you get), although my first impressions suggested that it is not as powerful in this one respect at *text87*. Having said that, users who find *text87* difficult to get to grips with will probably find *Perfection* a god-send.

One of the members, Terry Harman, has developed a plug-in eprom board for the QL. An

the microdrives to show all his *Toolkit 2* 'altkeys' and 'hotkeys', a very useful idea and one which I think I shall use myself. Yet



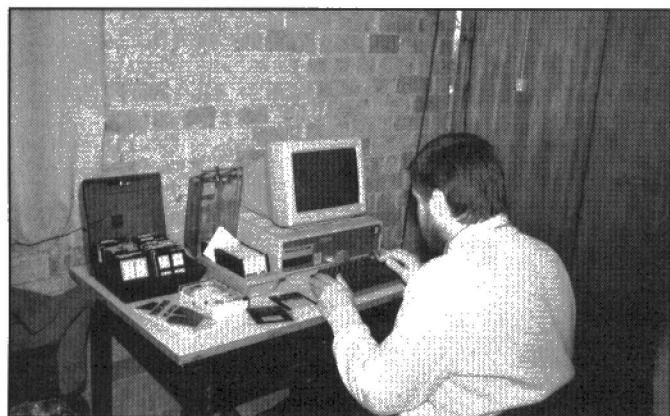
A discussion amongst the club members.

slips into the expansion port on the QL's left side. For users without an eprom programmer, Mr Harman is willing to 'blow' the eproms, with their chosen software on board, for them using QEP3. Unfortunately this device is not compatible with Miracle System's Trump Card, because that takes all the available memory space, leaving none for the eprom board, but it should nevertheless prove useful to a large number of QL owners.

Home boxed QLs were much in evidence at the March meeting. One member, Steve Bourne, has gone so far as to box his QL in an IBM personal computer case, complete with IBM keyboard (connected via a suitable interface), and three floppy drives. Another member used the original keyboard connected via a home-made extension cable complete with stickers above

another has his QL in a box, interfaced to two 10-megabyte hard disks. That particular size of hard disk is too small for PCs, because their programs are much larger than the QLs, but is perfect for the QL and can be purchased quite cheaply second hand.

There is quite a lot of interest in the Mandelbrot Set and other related fractals at QMAS, with members regularly bringing along new and faster programs. One member, Carl Cronin, has developed a Mandelbrot program that, instead of plotting the Mandelbrot Set on the screen as most programs do, prints it in eight colours on a Star LC10c printer. Using this technique full size A4 page Mandelbrots, with resolutions of 960 by 792 pixels can be generated and this method certainly lends itself to producing spectacular graphics. This program is public do-



One user's home-boxed QL set-up.

This diversity of membership ensures that a new member is bound to find people of similar ability and there are always people at a meeting to learn new things from and to answer any questions and solve any problems which a QL user is likely to encounter, from changing a keyboard membrane and getting a pound sign to print out, to learning a new computer language. Typical meetings are usually very busy, with many separate activities and discussions occurring simultaneously, and if the discussions go over your head, there is always the coffee and biscuits to resort to!

March's meeting was very active, with many new develop-

eprom (erasable programmable read only memory) is, as its name suggests, a type of computer memory that can, once written to with an eprom programmer, keep its contents even when the computer's power is switched off, until it is erased using a special ultra-violet eraser. This makes it very useful to computer users, because they can store their most commonly used programs and resident extensions in the eproms, which will then run as soon as the computer is switched on, without taking up valuable ram space.

Terry Harman's board, which can contain up to 192KB of program code in eproms, simply



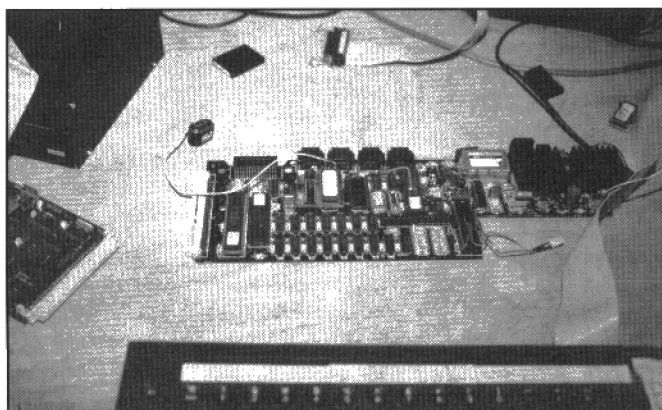
Jonathan Oakley of QView, with his robot.

main. It is available from CGH Services and in the near future it will be in the Quanta library.

QView, the developers of *Minerva*, are members of QMAS and had an exciting new development for QL users which they demonstrated. Having persuaded many QL users to rip the top off their machine and replace the native Sinclair JS, JM or even AH Qdos operating system with their derivative of it, *Minerva*, they have now developed an eeprom board. An eeprom (electrically erasable programmable read only memory) is similar to both a ram chip and the eprom chip explained earlier. Like an eprom, an eeprom can have data stored in it, which will remain even if the power is removed, but it does not need to be blown using a special eprom programmer; the QL can actually blow the chip itself.

Eeprom

The QView eeprom board, which sits in one of the rom sockets inside the QL, contains 255 bytes of eeprom and a real



The QView eeprom board, inside the QL.

time clock, and comes complete with *Minerva* version 1.88. The eeprom is used to hold the default settings for the QL, so that it remembers whether to boot into screen mode 4 or 8 when being switched on, and it knows which device to boot off. Lots of other useful pieces of information can be stored, such as a message from the user that will appear on the QL's screen before F1 is pressed, when it is switched on, or default settings for a word processor, or a password, to give some examples.

Jonathan Oakley, a member of QView and an ex-Sinclair em-

ployee, spent much of the evening demonstrating a new method he has developed of controlling robots and other similar devices from the QL. The eeprom device also has a system bus, which can be utilised, using a short piece of SuperBASIC and a small circuit, by would-be robot developers to send signals to their robot. Jonathan himself was using the bus to control an arm robot which was solving the Tower of Hanoi puzzle.

The QMAS Club itself has many resources available for loan to members, including the

complete Quanta program library consisting of over forty disks and a backup QL, for emergencies when a member's QL breaks down. The Club Chairman Peter Rowell is also thinking of purchasing, using Club funds, a ribbon re-inking machine, to allow members to re-ink their printer ribbons for as little as 50p.

Meetings

As I said at the beginning of this article, there are many QL clubs all round the country waiting for new members to drop in. Having been to a number of these meetings, I can say that they are very worthwhile. It is certainly worth getting in touch with the secretary of Quanta to see if there is a club near you. You don't have to be a member of Quanta to go along and I am sure that you too will find much of interest waiting for you.

For details of QUANTA subgroups and workshops, contact the Quanta General Secretary, Ron Durret, 38 Brunwin Road, Rayne, Braintree, Essex, CM7 5BU.

KEN BAKER SOFTWARE

32 Hunts Pond Road – Park Gate – Southampton – SO3 6QA – 0489 581056

All Programs for 768K Expanded QL with Trumpcard

SPEEDSORT

The ultimate sorter! Sorts arrays of any size by given element number. The speed is devastating – up to 80,000 sorts per second!

Machine Code £20.00

QLDL

QL Disk Library allows the user to catalogue up to 3000 program titles (around 75 disks at a time) in alphabetical order and cross-referenced by numbers given to the disks. No more searching for endless hours for that lost program.

Machine Code £15.00

SCREENMASTER

Screenmaster will record anything currently on screen in a 'squeezed' form, and reproduce the picture in a variety of cinematographic-style wipes and speeds.

Machine Code £20.00

PLANNER

One of the most useful programs you could wish for. It will provide a printout of dates, with day and week numbers, in a month-per-page format from a given starting date and number of forward days. It also includes routines for finding day and week numbers you can use in your own programs.

Machine Code £10.00

TRANQUIL

Tranquil converts Quill documents into BASIC PRINT statement form in order that they can be more easily manipulated for printing and adding printer instructions etc.

Machine Code £10.00

ALL PROGRAMS SUPPLIED ON TOP-QUALITY 3 1/2 INCH DISKS – POSTAGE & PACKING FREE

SOFTWARE FILE

INFORMATION

Program: Professional Publisher Toolbox
Price: £29.95 (plus 5/10% for overseas orders)
Supplier: Digital Precision Ltd.,
 222 The Avenue,
 Chingford,
 London E4 9SE.
 Tel: (081)-527-5493

PROFESSIONAL PUBLISHER TOOLBOX

Desktop publishing is a good area for developers of utilities.

The print trade uses a vast quantity of typefaces and enhancements, shapes, pictures, colours etc., and it is unlikely any dtp program will provide the serious user with more than his/her basic requirements when it is first introduced. Digital Precision's *Professional Publisher* is by no means in its first version, having been developed from their first essay into dtp - *Desktop Publisher* - and having had a variety of improvements made since its introduction (for example, the much quicker loading process of the latest version).

Defined

Comprehensive as ProPub is, there are several ways in which a developer can extend its range, perhaps the most obvious one being the provision of better-defined characters, particularly in large sizes. *DeLuxe Font Enlarger* (also from DP) deals with this area, but *Professional Publisher Toolbox* goes a bit further. It supplies additional QLS (standard) and HDF (high-definition) founts, and also provides tools for improving text format, importing non-DP clip art, and saving ProPub pages as screens which can be loaded into graphics programs.

The Toolbox is supplied only on disk, is for systems with at least 256 KB ram (ProPub itself needs this much), and is suitable for Thors, including the XVI. The supplied disk contained no less than 92 files - 36

Bryan Davies samples a new support bundle for the popular DP desktop publishing package.

QLS founts, 44 HDF founts, six screens, one page, a boot, three utility programs, and an Xtras extensions file. The boot is there solely to load the extensions file and will not be necessary if the user already has an appropriate version of DP's Xtras file (or equivalent) loaded for use with another program (eg *The Editor*, text 87, *Perfection*), to obtain the desired layout (text files loaded directly into ProPub may lose their paragraph splits, for instance).

The only instructions supplied with the view disk were on a single sheet of paper. The brevity of the information should not be a handicap for

experienced users; it is fairly obvious how to use the utilities once they have been EXEC-ed, and founts are simply loaded into ProPub in the normal way.

Pretext removes some oddities that can appear when Ascii text files are loaded into ProPub. Linefeeds are replaced by spaces, and nulls-CHR\$(0) - are then replaced by linefeeds. The effect is best illustrated by example, and the first illustration is of a text file Export-ed from *Perfection* and loaded directly into ProPub with the Configuration and Line Feed settings at WordWrap and Included, respectively. Note the 'half-lines'

and lack of paragraphs. The second illustration shows the same file loaded, with the same settings, after it had been processed by *Pretext*. The text format is now more sensible. As an incidental feature, this utility can be used on files from *The Editor* or *Perfection* before they are Imported into Quill, in the 'by paragraph' mode.

The *Grab Sector Clip Art* utility does what you would expect. It loads one of the Sector Software clip art files (designed to be used with *Page Designer*) to the screen and converts it into a form suitable for loading into *Professional Publisher*; that is, into a 32 KB

An imported file, pre-Pretext.

PERFECTION - THE NEW WORD-PROCESSING PROGRAM

Developing a word-processing program is no mean undertaking. Many people start on the task -- and soon abandon it. Even a "simple" program like Quill required a massive programming effort. All the time you are thinking about the project, other people are beavering away, doing the same thing. Worse still, they may introduce their "baby" before you are ready with yours, and there may have features you hadn't thought of, or which work better than ones you have written.

Fortunately for QL programmers, there is little competition. It was unlikely that anyone else would come out with a good word-processing program whilst *Perfection* was being developed, although there was a distinct possibility that further refinements of text87 would appear (the DeskJet & LaserJet printer-drivers, for example). The problem was

more one of deciding upon what features were needed to attract existing Quill users, and of incorporating them into a program of acceptable size. The more features are added, the greater becomes the testing requirement, and the harder it is to find and fix problems. Although there may be no-one else writing a WP program to compete with you, there may be someone writing another program that will cause some conflict with yours. A good example is the Minerva ROM, which has caused hiccups for several of D.P.'s existing programs. A less-likely occurrence is the discovery of an "unknown" feature of one of the Sinclair QDOS versions; by now, most of the oddities of AH, JM, JS etc have (presumably) been discovered, but another one did show itself during beta testing of *Perfection*. Fortunately, one of the Minerva developers was aware of the difference between JM and JS

The same file, post-Pretext.

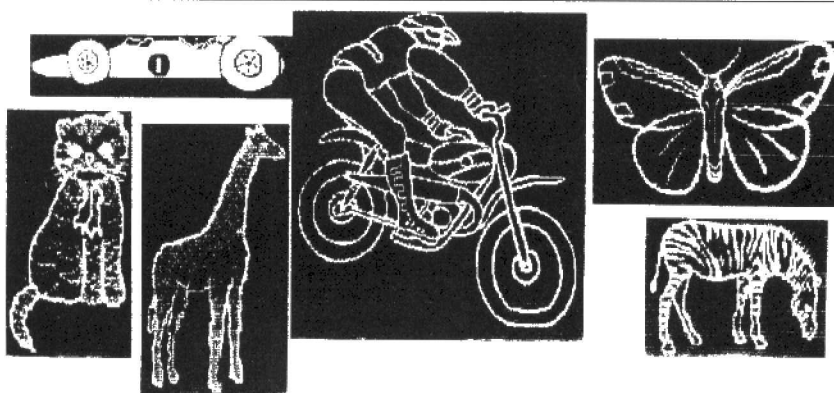
PERFECTION - THE NEW WORD-PROCESSING PROGRAM

Developing a word-processing program is no mean undertaking. Many people start on the task -- and soon abandon it. Even a "simple" program like Quill required a massive programming effort. All the time you are thinking about the project, other people are beavering away, doing the same thing. Worse still, they may introduce their "baby" before you are ready with yours, and there may have features you hadn't thought of, or which work better than ones you have written.

Fortunately for QL programmers, there is little competition. It was unlikely that anyone else would come out with a good word-processing program whilst *Perfection* was being developed, although there was a distinct possibility that further refinements of text87 would appear (the DeskJet & LaserJet printer-drivers, for example). The problem was more one of deciding upon what features were needed to attract existing Quill users, and of incorporating them

into a program of acceptable size. The more features are added, the greater becomes the testing requirement, and the harder it is to find and fix problems. Although there may be no-one else writing a WP program to compete with you, there may be someone writing another program that will cause some conflict with yours. A good example is the Minerva ROM, which has caused hiccups for several of D.P.'s existing programs. A less-likely occurrence is the discovery of an "unknown" feature of one of the Sinclair QDOS versions; by now, most of the oddities of AH, JM, JS etc have (presumably) been discovered, but another one did show itself during beta testing of *Perfection*. Fortunately, one of the Minerva developers was aware of the difference between JM and JS ROM versions which brought about this particular problem, and was able to point D.P. in the right direction (and thereby somewhat mollify D.P., after the aggravation over conflicts with Minerva). Users may differ on what is the biggest drawback of Quill.

Some Sector clip art grabbed and imported.



SBYTES file. The conversion is done quickly, with the on-screen prompts being sufficient explanation of what to do. The routine is less than helpful when the first conversion has been completed – it simply writes 'Done.' to the screen, and leaves the user with

no cursor, and no suggestions as to what to do next. Pressing ENTER achieves nothing apparent, and it seems necessary to return to SuperBasic and restart the routine. It is also necessary to do a separate SB directory of the input device, and list the required files

manually, since there is no directory function built-into the Grab routine itself. Not very helpful if you want to convert a few dozen files. The third illustration shows some Sector clip art, converted by Grab and loaded onto a ProPub page.

The **Screen From Page** routine will take part of an existing Professional Publisher page and save it as a screen (picture), in the form of a 32 KB SBYTES file. This allows that part of the screen to be loaded into a graphics program and modified; it can subsequently be reloaded into ProPub. The routine does not itself allow selection of part of a page – that has to be done in ProPub, by identifying the page section required there and saving it as a part-page.

Surplus

Alternatively, you can use the whole page, and allow the routine to deal with just the top, left part of it, assuming what you want is contained within that area. Any surplus material can be removed later, using a graphics program or ProPub. The **Screen** routine converts all of the chosen area into a 32 KB screen file. It doesn't care whether the page section concerned is graphics or text. The fourth illustration is of a combination of parts of two ProPub pages. The top section (the portrait) comes from one of Digital Precision's sample page files, and is actually a drawing done with the graphics program Eye-Q and imported into ProPub. The bottom section comes from a page created during the course of writing an article on using

ProPub, and is the top left-hand corner of a reproduction of the front cover of *QL World*, created directly with ProPub. The two parts of the illustration were converted separately with the **Screen** routine, then loaded as SBYTES pictures into ProPub, and printed from there, in a three-pass print of the resulting page on a nine-pin printer. This routine, also, ends one conversion operation with the message 'Done.', and you appear to have to restart the routine to convert another file; likewise, you need to do a directory beforehand, to know what file name you need for input.

**THIS IS THE
STONWORK5
HDF FOUNT**

and this is the SHARPI font.
Both of them are supplied on
the PROPUB TOOLBOX disk.

Two font samples

There are too many additional founts on the Toolbox disk to do more than show a couple of samples here. A total of 36 QLS and 44 HDF (high-definition) founts are supplied; two of the latter are shown in the fifth illustration. The space taken by these founts varies quite a bit, and the user may have to remove several of the default founts to get certain new ones in. For example, the STONWORK5 HDF fount is in a file of 16 KB, which is much larger than most of the other founts.

The three routines, and the founts, on the disk do their job well. They are all useful accessories for the Professional Publisher user. The less-experienced user would welcome more written advice on how to make use of the functions, or integration of the routines with ProPub itself, but it is fairly usual with such add-ons for them to be rather spartan in this respect. The results speak for themselves, as shown in the illustrations here. The user gains significant flexibility, without losing much time, and the cost is modest.



Learn to Love Your PRINTER

The first part of this article gave comments on making the basic entries in the table displayed by the *Quill* Install_bas program. Those are the entries which are relatively fixed, regardless of printer. This final part deals with the table entries which can be used to good effect to control *your* particular printer.

The Preamble and Postamble code settings hold the basic design of your sheet of printed paper; if any particular printer feature is going to be needed throughout printing, it should be called in the Preamble. For example, if you normally print in 12-pitch, switch that feature on with the Preamble. This doesn't stop you changing to any other pitch within a document, but you know that the printer will always start in 12-pitch mode. Likewise, if you normally use the USA standard character set, switch it on with the Preamble. The Postamble enables you to switch the printer back to some basic setting, when a print run has finished. You then know what it is going to be set to do when you next use it (assuming it has not been switched off); this is essential if you print from different sources – for example, Quill and SuperBasic – during one session, as it avoids the messy effect that the different source formats can produce. For example, if you end up a Quill document with condensed print, and then go on to list a SB program, the printer is not to know you don't still want condensed print, if you don't tell it so.

Entering Codes

All the entries from Preamble on down require a particular form of presentation, to meet the requirements of Install_bas. You have a choice of four ways of entering the character codes, but it may be that neither of these ways is exactly the same as that in which the codes are set out in your printer manual. Nevertheless, you must use the Quill way.

Codes have to be entered singly, separated by commas, with a maximum of 10 codes per entry. It is acceptable to mix the code types on a line. The types are: Ascii mnemonics, such as ESC, NUL, SOH; hexadecimal codes between \$0 and \$FF; decimal codes between 0 and 225; single characters preceded by a single or double quote mark, eg "I", '1', "4. For example, my printer manual states that the coding for setting Elite typestyle (12-pitch) is ESC "M", <1B>H<4D>H, or <27>10<77>10; this can be entered as ESC."M, or 27,77, or 1B,4D, or ESC,77, or 27,4D, or any other similar combination.

Returning to the Preamble sequence, it can be used to switch on alternative/extra character sets (you may want Greek, Spanish or other language symbols), margins, tab intervals etc.

The Postamble may not be needed, but a typical sequence is ESC,"@ to return the printer to the same settings as when it is first switched on. The Bold, Underline, Subscript and Superscript on/off entries are for the embedded Quill typestyle functions – those you can obtain using the F4 key. The entries are the codes your printer expects to see for the corresponding functions, such as ESC,"G for double-strike printing, which is a form of Bold print.

Check the descriptions of function actions carefully before using them; some functions do not work with others, a typical case being that emphasized print (another form of Bold) may not work with the Elite character size. Subscript and Superscript are functions which many users will never use and they can be converted to other functions. There is nothing to stop you inserting the printer codes for, say, italic print into the Subscript entry; then, whenever you use F4 L, you will get italic print. The screen presentation will still be the stylised, raised characters, however.

Many printers have more useful functions than the Quill printer driver can

Part Two

Bryan Davies surveys Quill commands to get the best from your printer.

command, and adding another two possibilities may be sufficient to solve a dilemma. The 10 Translate entries differ from the four embedded command entries in that you have to insert both the codes for the command keying and those for the corresponding printer function. This means that you have a maximum of nine codes left to describe to the printer what is required when one key is hit.

There may be more than one way of getting a particular printer function, and it can be necessary to look for the shortest coding. The first code is the one produced by the keyboard, and it must be in the range decimal 32 to 255. This limits you to using keyings that produce characters on the screen, which is desirable in that you have some indication of changes in print form.

If, like me, you want to keep keypresses to a minimum, you have to look at the keyboard and decide which characters shown there are not likely to be used within text; ~, @, : are examples which can be produced using two keys, and you may be able to do without one-key characters such as \, |,]. There is a wider choice if you accept three keys per character. All possibilities are listed in the 'character set and keys' section of the *QL User Guide*, and it is from there that you can get the corresponding codes to enter into the printer driver.

Sample table

A sample printer driver table is given here, for guidance only. The printer concerned is 'Epson-compatible', but that doesn't mean the codes are just the same as on an Epson. Look at this table for ideas on how to tackle producing your own particular requirements. Note that the Translates can be used both to switch on printer functions and to convert character codes from those used by the keyboard to those required by the printer for the same characters. The obvious example of the latter is the £ sign, so often printed out as something else. The QL keyboard code for this character is decimal 96; the same code produces a single quotation mark on the Kaga printer, with either the USA or the UK character set. To be able to print the # character, the USA set can be used; to get the £ as well (without using the code 35 needed for the #), an additional character set has to be activated.

The code 6 in the additional set on the

Kaga produces the £, so the Translate entry is simply 96,6. Here, the additional character set is switched on with the Preamble, but it could be switched on by the Translate 96,ESC,I,SOH,6. Each Translate entry can be used to produce only one printed character; it doesn't matter how many codes (up to nine) are used to specify that character, however. This is used in producing the Ø character; CTRL-SHIFT-R puts it onto the screen, and to get it printed the Danish character set is invoked (for this one character only).

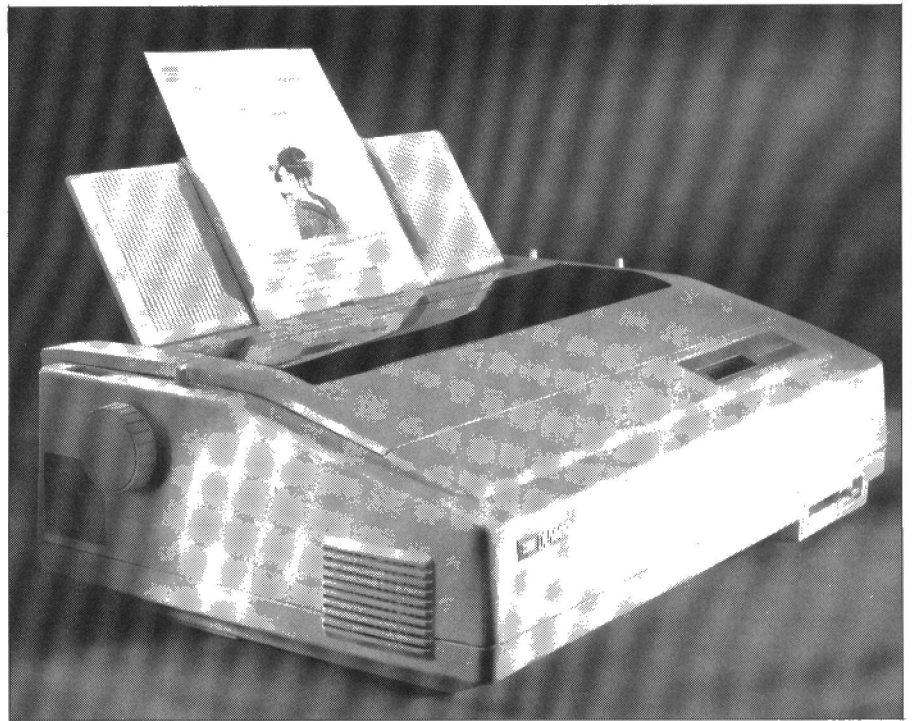
The Translate entry is 178,27,82,4,92,27,82,0 which is the QL code for Ø (178), followed by switching on the Danish set, followed by the Kaga code \ (92), then switching off the Danish set.

There are some points to remember when calling up typestyles. If you want to use several pitches, and features like proportional spacing, you have to ensure that switching off a feature returns the printer to a suitable state for what it has to do next. NLQ print is 10-pitch and, if your basic typestyle is 12-pitch, you have to switch off NLQ by telling the printer to switch to 12-pitch. The same applies to enlarged (and enlarged-condensed) Pica.

In the sample driver, the Preamble sets 12-pitch as the basic style (ESC,"M). NLQ is called by the ^ character (code 94), and switched off by calling Elite (12-pitch) again, hence Translate 2 reads ^,ESC,(ESC,p,SOH. Note here that you **cannot** type in ^ when entering this line – Quill wants the code itself. ESC,p,SOH is one way of specifying 12-pitch, and ESC,"M works just as well. Proportional spacing may work only with 10-pitch, and has to be switched off deliberately when 12-pitch is returned to.

One easy trap to fall into is the one set by the printer makers when using three or more codes for a function. The third code may be shown as simply n, and it might be less than obvious what form this has to take – decimal, straight numeric, etc. Where NUL or SOH appear in the sample driver, you have examples of n being set to 0 or 1. The actual entry typed in to produce the screen entry ESC,R,NUL was 27,82,0. On the other hand, the ESC " " + n function has a list of 64 typestyles (on the Kaga) and they cannot be entered by using the numbers 0-63 as Ascii codes (that is after a quotation mark); to set bold, enlarged-condensed Pica, you can enter ESC,"1,"4, or 27,33,52 etc. 52 is the listed decimal number, 4 is the equivalent Ascii character (that appears on the screen when you hit the 4 key).

The decimal number is arrived at by looking at a table of (computer) bit settings, and selecting the features required; in this case enlarged print requires the fifth bit to be set 'on', double-strike (bold) requires the fourth bit to be on, condensed requires the second bit to be on. This means that the decimal code (the sum of the bit settings) is $1 \times 2^5 + 1 \times 2^4 + 1 \times 2^2 = 52$. The combination of enlarged and con-



densed Pica give 8.55 characters/inch, a useful medium-size print for headings.

The purpose of Translate 7 may not be obvious. In order to maximize the available Translate entries, this one setting is used to switch off any or all of the others. By doing this, you don't have to use two entries for each function, one each to switch it on and off. Translate 7 sets the print style back to Elite (12-pitch), turns proportional spacing off, and turns italics off.

Turn off and on

Using this approach, you have to remember that the 'off' keying will turn off features you may want to continue using, and it may be necessary to turn them back on immediately afterwards. If you have chosen to have both condensed and italics on, together, then to switch off only the italics, the 'off' switch puts both italics and condensed out of action, and you have to reset condensed. This situation should not occur frequently, and the need to reset a feature occasionally is a small price to pay for the ability to have several more Translate entries available, and for not having to remember a variety of different keys for switching features off. Of course, however you look at it, having to use 'odd' symbols and key combinations to select printer features is far from being the ideal way of making the features easy to use, but we have to live within the constraints imposed upon us by the software and hardware designers. I much prefer to use mnemonic keying, but that really isn't feasible with the basic Quill; when dealing with *The Editor* later, I will explain how more-recognisable key combinations can be set up to call the features.

It is surprising that, when Quill was re-

written for use on PCs, the number of embedded feature-calls was not increased much; the only additional feature available is italic print, but even that puts Quill well ahead of many programs costing several hundred pounds. For those using both a PC and a QL – maybe the former in the office, and the latter at home – it is quite straightforward to set the respective printer drivers up to use essentially the same key combinations on both machines, with only minor variations to allow for the different way italics have to be called and for keyboard variations.

A sample printer driver for Kaga-Taxan KP-810 (Canon PW1080A) printer follows. The basic setting, from the Preamble entry, is Elite (12-pitch) type on, Printable Control Code range on, Printable Range Expansion on. The embedded Quill features F4 B/U/H/L are set to give those features – Bold/Underlined/Super-script/Subscript. The Translate entries 1-10 call these features:

- (1) proportional Pica (10-pitch; proportional spacing is always bold),
- (2) proportional nlq (bold and 10-pitch),
- (3) enlarged, proportional Pica (bold, 5-pitch),
- (4) condensed Pica (17.14-pitch),
- (5) enlarged-condensed bold Pica (8.55-pitch),
- (6) enlarged, bold Elite (6-pitch),
- (7) italics and proportional spacing off, Elite on ('everything off'),
- (8) convert keyboard £ to printer £,
- (9) italics on,
- (10) convert keyboard Ø to printer Ø.

In a few cases, combinations of calling codes may not produce the desired result unless they are in a certain order; in the examples given here, it is necessary to key

LEARN TO LOVE YOUR PRINTER

Screen display:

NAME KP-810
PORT SER1
BAUD RATE 9600

PARITY NONE

LINES/PAGE 70
CHARACTERS/LINE 134
CONTINUOUS FORMS NO

END OF LINE CODE CR,LF

PREAMBLE CODE ESC,M,ESC,I,SOH,ESC,6 27,77,27,73,1,27,54

POSTAMBLE CODE ESC,@ 27,64

BOLD ON ESC,G 27,71

BOLD OFF ESC,H 27,72

UNDERLINE ON ESC,-,SOH 27,45,1

UNDERLINE OFF ESC,-,NUL 27,45,0

SUBSCRIPT ON ESC,S,SOH 27,83,1

SUBSCRIPT OFF ESC,T 27,84

SUPERSCRIP ON ESC,S,NUL 27,83,0

SUPERSCRIP OFF ESC,T 27,84

TRANSLATE 1 @,ESC,!,NUL,ESC,p,SOH 64,27,33,0,27,112,1

TRANSLATE 2 ^,ESC,(,ESC,p,SOH 94,27,40,27,112,1

TRANSLATE 3 !,ESC,!,0,ESC,p,SOH 124,27,33,48,27,112,1

TRANSLATE 4 {,ESC,!,EOT 123,27,33,4

TRANSLATE 5 },ESC,!,4 125,27,33,52

TRANSLATE 6 ~,ESC,!,1 126,27,33,49

TRANSLATE 7 ©,ESC,5,ESC,!,SOH,ESC,p,NUL 127,27,53,27,33,1,27,112,0

TRANSLATE 8 £,ACK 96,6

TRANSLATE 9 \,ESC,4 92,27,52

TRANSLATE 10 Ø,ESC,R,EOT,\,ESC,R,NUL 178,27,82,4,92,27,82,0

Comments/decimal code keying:

(10 characters maximum)
(only alternative is SER2)
(or 75/300/600/1200/
2400/4800)
(or SPACE/MARK/ODD/
EVEN)
(0 to 255)
(0 to 255)
(or YES)

switching command. These combinations have to be entered as CTRL-SHIFT-alpha. If you don't like doing this, try choosing another mnemonic for the feature concerned; use CTRL-S ('small') for condensed print, rather than CTRL-SHIFT-C, for example.

Choice of method

The sample Driver_dat file has much the same features in it as the sample Quill Printer_dat file, as would be expected since the aim is to get the same printed output. Like Install_bas, you can enter the codes in various ways, your choice of method being influenced by personal familiarity with code systems as well as the presentation in the printer manual. I find decimal the least complex; the only difference from Quill here is that the # prefix has to be used with all decimal codes.

The separating comma between codes is mandatory, as before.

All lines giving instructions to the print program (Edtprt bin) must start with a semi-colon; lines which are for information purposes only – equivalent to SuperBasic REMark lines – must start with a period/full stop.

There are no serious restrictions on how many instructions are passed to the printer, and the sample includes a set of code-changes for printing out characters from other European languages. If you check the keyboard codes given for such characters in the QL User Guide list against the characters produced on the screen, you may notice they do not all agree, and some experimentation may be required. The entries are all PXL (affecting one line only) because they are to take effect only at the point the code is met in the text. This avoids strange hangover effects in subsequent printing. The entries for typestyle changes are all PXG (global) in effect. They can be compared directly with those in the sample Printer_dat file. One slight difference between Quill and The Editor is that combining a Quill embedded command with a Translate command does not always produce the desired result, whereas all the Editor commands are of the Translate form and can be combined as required. The Bold feature was incorporated into other feature calls in Printer_dat, as using F4-B with, say,) for medium bold Pica does not seem to work.

Headers and footers

The margins for page headers and footers (PHM and PFM) are set to give a total of 70 lines when added to the text length, PPL. You can re-specify these parameters on individual pages of a document, but don't change just one without making a balancing change to one or both of the others. That is, the basic text length of 60 lines could be changed to, say, 63, to accommodate one page where it is essential to get certain text onto that page rather than the next one, but

;^, rather than ^:, to get enlarged proportional nlq.

Reverting back to the point about non-printing codes spoiling the line justification, one way of partially fixing the problem is to insert a space (or two) into the Translate entries concerned. Take italics in the sample above; as it stands, putting the character \ into a document will cause the printer to use italic type, and to shorten the line it is on by one space for every occurrence of the \. If the code conversion for switching off all codes is changed to 127,27,53,27,33,1,27,112,0,32, a space will be 'printed' each time any of the typestyle codes is switched off. The print begins to look rather strange if you insert more than one space each time, however, so that it is not feasible to balance out all codes.

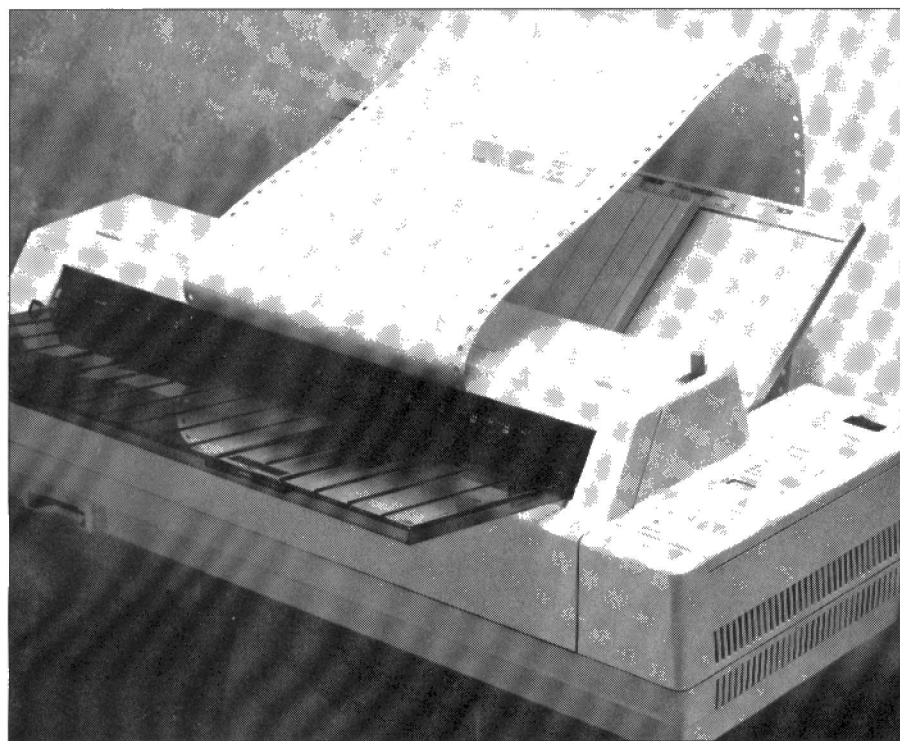
Flexability

To Create a Driver_dat file for The Editor, the process is essentially similar to that with Printer_dat in Quill, but there is much more flexibility to choose what you want to do with the printer. The greater range of possibilities offered does entail doing more thinking, if you are to really use the features, but you can make a minimal file without great effort. It is helpful to have gone through all the agonies with Quill beforehand – not that this will necessarily

prevent you making the same boobs! The file can be created in The Editor itself, and there is a sample file to give guidance. The manual goes into much more detail than the QL User Guide, and the comment here is on only those points where I had trouble.

It is possible to make the keying compatible with that used in Quill, but it is likely you will end up not doing this, for various reasons. Now you have the opportunity to use control codes which do not involve 'losing' normal keyboard characters, you may choose to do so. Being able to call more printer features means finding more key combinations, and most people won't be able to spare any more of the normal set. The chance to use mnemonic representations of the printer features is not to be missed. The Editor has its own fount of characters, produced by pressing CTRL before alpha keys; the screen character is the normal alpha one, with a bar over the top (overscore or overline). CTRL-B produces B on the screen; you can find the corresponding decimal or hexadecimal code in the character set and keys list in the QL User Guide.

For most purposes, then, you need only two keys for any printer feature to be called. Unfortunately, some combinations of CTRL and alpha have operating system definitions, and cannot be used. An obvious example is CTRL-C, which is the task-



there would have to be a corresponding reduction of 3 in PHM or PFM, or split between the two of them. PHM is best left at 2, or 1 at the minimum, to allow the printer time to get the paper moving precisely, before it prints the first few lines of text.

Sample Driver_dat file for Kaga-Taxan KP-810 (Canon PW1080a) printer

.18 May 88:Driver_dat
;Name KP-810

```
;PDV ser1
;PBD 9600
;PPA #27,#54,#27,#77,#27,#73,#1
;PPO #27,#64
;PHM 2
;PFM 8
;PPL 60
;PLM 8
;PEL #10
;PSS
;PXL £;#96;#6
;PXL Ø;#178;#27,#82,#4,#92,#27,#82,#0
;PXL à;#128;#27,#82,#2,#123,#27,#82,#0
;PXL ô;#132;#27,#82,#2,#124,#27,#82,#0
;PXL ü;#135;#27,#82,#2,#125,#27,#82,#0
;PXL Å;#160;#27,#82,#2,#91,#27,#82,#0
;PXL Ö;#164;#27,#82,#2,#92,#27,#82,#0
;PXL Ü;#167;#27,#82,#2,#93,#27,#82,#0
;PXL ß;#156;#27,#82,#2,#126,#27,#82,#0
;PXL è;#144;#27,#82,#1,#125,#27,#82,#0
;PXL é;#131;#27,#82,#1,#123,#27,#82,#0
;PXL à;#141;#27,#82,#1,#64,#27,#82,#0
;PXL ç;#136;#27,#82,#1,#92,#27,#82,#0
;PXL ù;#154;#27,#82,#1,#124,#27,#82,#0
;PXL °;#186;#27,#82,#1,#91,#27,#82,#0
;PXL §;#182;#27,#82,#1,#93,#27,#82,#0
;PXL Pt;#157;#27,#82,#7,#35,#27,#82,#0
;PXL ;;#179;#27,#82,#7,#91,#27,#82,#0
;PXL N;#181;#27,#82,#7,#92,#27,#82,#0
;PXL ;;#180;#27,#82,#7,#93,#27,#82,#0
```

```
;PXL ñ;#137;#27,#82,#7,#124,#27,#82,#0
;PXG Double Strike;#2;#27,#71
;PXG Underline;#21;#27,#45,#1
;PXG Superscript;#8;#27,#83,#0
;PXG Subscript;#11;#27,#83,#1
;PXG Proportional NLQ;#14;#27,#40,
#27,#112,#1
;PXG Double Strike Proportional Enlarged
Pica;#7;#27,#33,#48,#27,#112,#1
;PXG Double Strike Enlarged Elite;#5;
#27,#33,#49
;PXG Condensed Pica;#19;#27,#33,#4
;PXG Medium Pica;#13;#27,#33,#36
;PXG Proportional Pica;#16;#27,#33,#0,
#27,#112,#1
;PXG Italics;#20;#27,#52
;PXG All off;#15;#27,#53,#27,#33,#1,#27,
#84,#27,#45,#0,#27,#72,#27,#112,#0
```

The keyings mostly have some mnemonic content – CTRL-B for B, CTRL-S for small (condensed), CTRL-H for high (superscript). CTRL-I is used to indicate the end of a page, so is unavailable for low (subscript); CTRL-K has been used, as being the nearest, alphabetically, to H (for high). To distinguish between the two enlarged types, CTRL-G has been used for Pica (giant – because it is the largest available) and CTRL-E for Elite (enlarged). CTRL-M suits medium (enlarged-condensed). CTRL-I is a system code, so CTRL-T has been used for Italics. Hopefully, suggesting such key combinations will have the effect of goading users into generating their own, better ones!

Several methods of introducing code translations have been suggested by readers over the past year, and some of them may suit individual users better than the method given in Part 1 of this article. For instance, the ESC code needed as a preliminary to get a printer to change most

features can be entered into a document by typing CTRL-;, which appears on the screen as û. To make the keying clear – hold down the CTRL key and tap the ; key. These are the keys which produce the decimal code 155. The printer converts this code to the one for ESC. So, when you want italics, type û"4" into your text (your printer may require the second code in a somewhat-different format). My own feeling is that the method described in this article is more appropriate for users who want the selection of features to be as automatic as possible, but the method just mentioned may be preferable for users with more knowledge of the mechanism of getting printed output.

Easy and difficult

There are two other programs to consider when talking of printer drivers. *text⁸⁷* makes life both easy and difficult. As an ordinary user, there is nothing you can do to alter the supplied drivers, and that makes life much easier than it is with Quill or The Editor. Should you have a real need to change a supplied driver, you would have to either ask the suppliers of the program to write it for you, or ask for the assembly code of the driver and make the changes to that yourself. For a user with sufficient programming knowledge, and having a suitable assembler program, making changes this way is straightforward. The other program is due on the scene any day now, at the time of writing – Perfection. The printer driver for this can be tailored by the ordinary user, but comments on doing this will have to wait until the print selection of the program has been finalised, and I've had time to work with it.

INFORMATION

Serial-parallel interface:
Miracle Systems Ltd.,
25 Broughton Way
Osbalwick
York YO1 3BG.
Tel (0904)-423986

The Editor:
Digital Precision Ltd.
222 The Avenue
Chingford
London E4 9SE.
Tel. (081) 527-5493

Text⁸⁷:
Software⁸⁷
33 Savernake Road
London NW3 2JU.
Tel. (071) 267-2025

Psion 2.35 programs:
T.K. Computerware
Stone Street
North Stanford
Ashford
Kent TN25 6DF.
Tel. (030381)-2801

SUBSCRIBE TO SINCLAIR QL WORLD AND GET THIS LOCKABLE DISK BANK FOR 40 3.5 INCH DISKS



...



ABSOLUTELY FREE!!

Please send me my 12/24 month subscription to SINCLAIR QL WORLD and my free disk bank.
I enclose my cheque/money order for £ made payable to MCPC Ltd or debit my Access/Visa card.

No. Expiry date

Name Address

(Please enter postcode to ensure prompt delivery)

Signed Date SQL 0791

UK	£23.40	12mth	EUROPE	£32.90	12mth	REST OF	£40.90	12mth
	£46.80	24mth		£65.80	24mth	WORLD	£81.80	24mth

OVERSEAS RATES INCLUDE AIRMAIL SERVICE

Fill in the coupon on the left and send it with your remittance to: MSM Subscriptions Department, Lazahold Ltd, P.O. Box 10, Roper Street, Pallion Industrial Estate, Sunderland SR4 6SN. (The first issue of a new subscription to be delivered will be one or two issues after the one you placed your order in.)

Offer Closes 31st July 1991.